Universidade de Aveiro Departamento de Engenharia Mecânica
2016

BERNARDETE DA COSTA COELHO

ENERGY EFFICIENCY OF WATER SUPPLY SYSTEMS USING OPTIMISATION TECHNIQUES AND MICRO-HYDROTURBINES

EFICIÊNCIA ENERGÉTICA DE SISTEMAS DE ABASTECIMENTO DE ÁGUA RECORRENDO A TÉCNICAS DE OPTIMIZAÇÃO E USO DE MICRO-HIDROTURBINAS

Universidade de Aveiro Departamento de Engenharia Mecânica
2016

BERNARDETE DA COSTA COELHO

ENERGY EFFICIENCY OF WATER SUPPLY SYSTEMS USING OPTIMISATION TECHNIQUES AND MICRO-HYDROTURBINES

EFICIÊNCIA ENERGÉTICA DE SISTEMAS DE ABASTECIMENTO DE ÁGUA RECORRENDO A TÉCNICAS DE OPTIMIZAÇÃO E USO DE MICRO-HIDROTURBINAS

Tese apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Doutora em Engenharia Mecânica, realizada sob a orientação científica de António Gil d'Orey de Andrade Campos, Professor Auxiliar do Departamento de Engenharia Mecânica da Universidade de Aveiro

Apoio financeiro da FCT e do FSE no âmbito do III Quadro Comunitário de Apoio.

To my strong mom Maria and my lovely brother Cláudio,
To Marlene, my shining star,
To Pedro, my everything :)

o júri / the jury

presidente / president

Doutor Aníbal Manuel de Oliveira Duarte
Professor Catedrático, Universidade de Aveiro

Doutora Maria da Conceição Morais de Oliveira Cunha
Professora Catedrática, Faculdade de Ciências e Tecnologia, Universidade de Coimbra

Doutora Dídia Isabel Cameira Covas
Professora Associada, Instituto Superior Técnico, Universidade de Lisboa

Doutor Almerindo Domingues Ferreira
Professor Auxiliar, Faculdade de Ciências e Tecnologia, Universidade de Coimbra

Doutor Hugo Miguel Filipe Calisto
Investigador, Joint Research Centre - Institute for Energy and Transport (JRC-IET), Itália

Doutor José de Jesus Figueiredo da Silva
Professor Auxiliar, Universidade de Aveiro

Doutor João Alexandre Dias de Oliveira
Professor Auxiliar, Universidade de Aveiro

Doutor António Gil d'Orey de Andrade Campos
Professor Auxiliar, Universidade de Aveiro (orientador)

agradecimentos /
acknowledgements

Everyone who somehow crossed in my life through the last four years, directly or indirectly contributed to this final result. I have to thank you all for the good things that sustained me and also for the less good things that made me learn and grow. Without contact with some particular institutions/companies/groups and without the support of some particular persons, this wouldn't have been as interesting, entertained and exciting to fulfil.

To my supervisor, Prof. Gil, who was always open to discussions and provided so many useful insights on this work, thank you for making this journey possible, for teaching me so much, for helping me in the moments I most needed, for being so optimistic and so patient, for resisting to my "kind of crazy" moments, and mostly, for being such a good supervisor!

To FCT (*Fundação para a Ciência e a Tecnologia*, Portugal), for the financial support with the grant SFRH/BD/82191/2011.

To the GRIDS research group, an incredible group and the best team I have ever been part.

To Telesensor, Lda., for the great challenge through the cooperation project with the University of Aveiro. The contact with the "real world" represented a crucial part in the development of this work.

To UNESCO-IHE, Institute for Water Education, thank you for the internship opportunity. More than knowledge, the amount of multi-cultural and multi-organizational contacts and experiences provided by such a place, allowed me to improve a large number of skills. I want to thank the professors of the hydroinformatics group, Prof. Ioana Popescu and Prof. Andreja Jonoski for receiving me so well, and especially Prof Dimitri Solomatine for the so interesting conversations about forecasting and optimisation and for all the good advices. I also want to thank all my colleagues from IHE, especially Juan Carlos and Gonzalo, for the friendship, for the great conversations and all the good moments.

To everyone I had the pleasure to met in COHiTEC, especially to the mentors and to the team responsible for this fantastic training programme. Thank you for providing me important business insights and knowledge, which significantly contributed to enrich my work. Thank you for the rewarding experience.

To the people that work in the Department of Mechanical Engineering, specially to Tânia, Cecília, Marisa and Carla, thank you for being always available to help me with everything and for being so kind.

To my “old” and current office mates, Joana, Marisa, Raquel, Elisabete, Ana, Patricia, Tiago and Zé, thank you for the great moments (even the ones characterised by a smell of nails varnish!), for the good advices, for the friendship, for all the jokes, for the problems shared and discussed, for the “special” emails, for the surprises, for the chocolates and cinnamon cakes, for being there, for keeping in touch! And to other friends that were also present in several of these and so many other great moments: Dulce, Mina, Margarida, Pedro, Laranjeiro, Diogo Cardoso, Joana, Abreu, Diogo, Sena, Miguel, Bastinhos, Nelson, Rui Paulo... You were equally part of this journey and I would like to thank you for all the special moments and conversations we had :)

To the MSc students that had the bad luck (or not!) of developing their dissertations under my mentoring, Bruno, Daniel, Joana and Mário, I would like to thank you for the productive discussions, for making me look at my work under different points of view, for letting me help you, for listening (even if not always...), for letting me grow with you!

To my great friends from Braga, especially to the “panisgas”, to the toys group, and to Filipa and Mariana (*pastoraaas*...), my friends for life. Thank you for all the amazing moments and, mostly, thank you for the support not only during the last four years but since ever...

To my crazy Vet friends from UTAD, thank you for everything that I should not mention here :), for teaching me something useful (or not?) about animals and for making me see other perspectives...

To three special friends, Teixeira-Dias, JAlex and Gil, who are inspiring persons to me, each one for multiple distinct characteristics... I want to thank you for influencing so positively my work and my life!

To my big family, specially to Maria, Cláudio, Olívia, Joaquim, Ana, Rui, Miguel, grandmother Sofia and grandparents Manita and Ângelo, thank you for making me who I am today... Thank you for making my life better... Thank you for caring... Thank you for being such a good family!

To my special one, Pedro, thank you for making me stronger, for never let me quit, for making me laugh in that stupid moments, for making me happy! Your support and companionship have been vital during the entire journey... You and our amazing pets, Betty, Pulguinha, Violeta and Lótus, make every day worth (even the less good ones!)...

It is more than appropriate to cite the popular quote “*If I have seen further it is only by standing on the shoulders of giants*” (Isaac Newton). Thank you all for letting me stand on your shoulders! Now I feel like a giant too!!!

keywords

demand forecasting, energy efficiency, energy recovery, hydraulic modelling, optimisation, water supply and distribution

abstract

Energy efficiency plays a large role in the sustainability effort of water utilities since, globally, 35% of the total expenses with water production (12 billion euros) are being spent on energy. The main obstacle for the efficiency improvement in water supply systems is mostly related to the complexity of the systems and also to the low levels of resilience in their operations.
The main goal of this thesis is the development of an automatic computational methodology capable of (i) applying distinct techniques for the optimal operation of any water network and (ii) searching for possible locations for energy recovery using turbines and then selecting the most adequate type of turbine. Four major topics that play crucial roles in the networks efficiency improvement are addressed: (i) modelling and simulation of the networks, (ii) operational optimisation, (iii) water demand forecasting and (iv) energy recovery using hydroturbines.
An optimisation approach dealing simultaneously with valves, fixed-speed pumps and variable-speed pumps is proposed. The operating periods of all valves and pumps and also the speed settings of variable-speed pumps are used as decision variables in an optimisation problem in order to minimise the energy costs associated to a network operation. The hydraulic simulator EPANET 2.0 is used to verify and ensure the feasibility of the obtained solutions. Benchmark networks are tested with distinct optimisation techniques including several algorithms, such as Nelder-Mead Simplex, Genetic Algorithms (GA), Particle Swarm Optimisation (PSO) and Differential Evolution (DE).
A novel formula for predicting the pumps efficiency changes with speed is proposed and compared with the few existing methods, including the one used by EPANET.
An automatic process for the analysis of any water network in order to locate sites and select turbines for energy recovery is implemented and validated with a case-study.
Finally, models for predicting short-term water demands are developed and tested with data collected from a Portuguese water network. Traditional forecasting models based on exponential smoothing and naïve models are developed using a spreadsheet while artificial neural network-based models are developed in Matlab. The effect of distinct input variables (including anthropic and meteorological variables) in the ANN-based models is analysed.

palavras-chave

abastecimento e distribuição de água, aproveitamento de energia, eficiência energética, modelação hidráulica, optimização, previsão de consumos

resumo

A eficiência energética representa um papel significativo no esforço para a sustentabilidade por parte das empresas das águas, uma vez que, mundialmente, 35% dos custos totais com a produção de água (12 mil milhões de euros) estão a ser gastos em energia. Os principais obstáculos à melhoria da eficiência energética dos sistemas de abastecimento de água estão essencialmente relacionados com a complexidade dos sistemas e também com os baixos níveis de resiliência a nível operacional.

O principal objectivo desta tese é o desenvolvimento de uma metodologia computacional automática capaz de: (i) aplicar diferentes técnicas para operação optimizada de qualquer rede de água, (ii) procurar por possíveis localizações para recuperação de energia utilizando turbinas e, posteriormente, seleccionar o tipo de turbina mais adequado.

Quatro tópicos fundamentais que representam um papel crucial na melhoria da eficiência energética das redes são abordados: (i) modelação e simulação das redes, (ii) optimização operacional, (iii) previsão de consumos de água e (iv) aproveitamento de energia utilizando hidroturbinas.

Uma abordagem de optimização para lidar simultaneamente com válvulas, bombas de velocidade fixa e bombas de velocidade variável é proposta. Os períodos de operação de todas as válvulas e bombas, assim como as velocidades das bombas de velocidade variável são utilizados como variáveis de decisão num problema de optimização de forma a minimizar os custos energéticos associados à operação de uma rede. O simulador hidráulico EPANET 2.0 é utilizado para verificar e garantir que as soluções obtidas são viáveis. Redes benchmark são testadas com diferentes técnicas de optimização, incluindo vários algoritmos, tais como Nelder-Mead Simplex, Algoritmos Genéticos (GA), optimização por bando de partículas e evolução diferencial.

Propõe-se uma nova formulação para prever a variação de curvas de eficiência das bombas com a velocidade e esta é comparada com os poucos métodos existentes para o efeito, incluindo o utilizado pelo EPANET.

Um processo automático para a análise de qualquer rede hidráulica de forma a identificar locais e seleccionar turbinas para aproveitamento de energia é implementado e validado com um caso de estudo.

Finalmente, modelos para previsão de consumos de água a curto prazo são desenvolvidos e testados com dados recolhidos de uma rede de água Portuguesa. Modelos de previsão tradicionais baseados em modelos de alisamento exponencial e modelos naïve são desenvolvidos utilizando folhas de cálculo, enquanto que os modelos baseados em redes neuronais artificiais são desenvolvidos no Matlab. Analisa-se ainda o efeito de diferentes variáveis de entrada (incluindo variáveis antrópicas e meteorológicas) nestes últimos modelos.

Contents

List of Figures

List of Tables

Nomenclature

f^{A}	-	activation function
α_{p}	-	amount of pump peak power occurrence
P_{u}	-	approximated capacity of an unit hydropower scheme
D_{a}	-	approximated turbine runner diameter
$\bar{y}$	-	arithmetic mean of observed variables
f_{L}	-	augmented Lagrangian function
ρ_{h}	-	autocorrelation coefficient at lag h
ϕ	-	autoregressive (AR) order (or AR lags)
B_{s}	-	back-shift operator
θ^{b}	-	bias node
k_{B}	-	Boltzman constant
$\mathbf{x}_0$	-	centroid (mean) of a simplex
$\mathrm{d}S/\mathrm{d}t$	-	change in storage volume in the time interval t
f_{civil}	-	civil cost factor
c_1 and c_2	-	cognitive and social learning rates
u^{c}, w^{c}	-	connection weights
k_1, k_2	-	constants
$\mathbf{x}_{\mathrm{c}}$	-	contracted point of a simplex
β_{c}	-	contraction coefficient
f_{c}	-	cooling factor
T_{c}	-	cooling temperature
$C_{\mathrm{demandCharge}}$	-	cost associated to the pumps operating at periods of peak demand
C_{civil}	-	cost of civil works
$C_{\mathrm{t,CF}}$	-	cost of Cross-flow turbine
$C_{\mathrm{t,F}}$	-	cost of Francis turbine
$C_{\mathrm{t,K}}$	-	cost of Kaplan turbine
$C_{\mathrm{t,P/T}}$	-	cost of Pelton/Turgo turbines
$C_{\mathrm{t,prop}}$	-	cost of propeller turbine
C_{pumping}	-	cost of pumps operation
η_{CF}	-	Cross-flow turbine efficiency
x_{scaled}	-	decision variable scaled in the range $[0, 1]$
$\mathrm{DC_p}$	-	demand charge per unit of power
υ	-	differencing order
C_Q	-	dimensionless flow

C_H	-	dimensionless head
C_{P_e}	-	dimensionless power
N_Q	-	dimensionless turbine specific speed based on flow
$\mathbf{x}^{G+1}$	-	disturbed (mutated) vector
η_t	-	efficiency of the set turbine, transformer, generator and gearbox
P_e	-	electric power
Z	-	elevation
C_e	-	emitter coefficient
ΔE_{level}	-	energy difference between two consecutive states
E_{level}	-	energy level
E	-	energy
h	-	equality constraint function
λ_h	-	equality Lagrange multiplier
ε	-	error/residual
$\hat{\beta}$	-	estimated regression coefficients
χ_{exceed}	-	exceedance probability
$\mathbf{x}_e$	-	expanded point of a simplex
γ_e	-	expansion coefficient
q	-	flow correction factor
b	-	flow rate exponent
Q_{ij}	-	flow rate from node i to node j
Q	-	flow rate
ρ	-	fluid density
γ	-	fluid specific weight
v	-	fluid velocity
$\hat{y}$	-	forecasted variable
V	-	fraction of energy losses dependent on Re
$\eta_{F,above}$	-	Francis turbine efficiency for flows above Q_{peak}
$\eta_{F,below}$	-	Francis turbine efficiency for flows below Q_{peak}
$\eta_{F,peak}$	-	Francis turbine peak efficiency
F^H	-	function of the equality constraint h and the corresponding penalty coefficient r_h
F^G	-	function of the inequality constraint g and the corresponding penalty coefficient r_g
η_{gear}	-	gearbox efficiency
f	-	general objective function (or fitness)
G	-	generation number
η_{geb}	-	generator efficiency
K_y	-	geometric mean of observed variables
$\mathbf{G}^{best}$	-	global best position
g	-	gravitational acceleration
H_{gross}	-	gross head
P_{gross}	-	gross power
C	-	Hazzen-Williams coefficient
H	-	head
h_m	-	headloss due to minor losses

J_{t}	-	higher cost vertical axis turbine factor
g	-	inequality constraint function
λ_{g}	-	inequality Lagrange multiplier
θ_{g}	-	inequality penalty coefficient
$\mathbf{B}$	-	inverse of the derivatives of the headloss in links
$\mathbf{A}$	-	Jacobian matrix of size $n_{\mathrm{nodes}} \times n_{\mathrm{nodes}}$
η_{K}	-	Kaplan turbine efficiency
$\eta_{\mathrm{K,peak}}$	-	Kaplan turbine peak efficiency
h	-	lag (h-step ahead)
L	-	lagged observation
L^{s}	-	level component/smoother
S^{s}	-	level component/smoother
K_{t}	-	lower cost vertical axis turbine factor
P	-	manometric pressure (or pressure head)
$\mathbf{AggregStep}$	-	matrix of aggregated steps
$\mathbf{Z}$	-	matrix of predictor variables
$\mathbf{F}$	-	matrix of size $n_{\mathrm{nodes}} \times 1$
$\mathbf{X}_{\mathrm{opt}}$	-	matrix of the decision variables obtained in the optimisation process
$\mathbf{X}$	-	matrix of the decision variables
$\mathbf{X}_0$	-	matrix of the initial decision variables
$P_{\mathrm{e,max}}$	-	maximum value of pump power needed
L_{max}	-	maximum water level of a tank
P_{min}	-	minimum manometric/head pressure required
L_{min}	-	minimum water level of a tank
$x_{\mathrm{min}}/x_{\mathrm{max}}$	-	minimum/maximum values of $x_{\text{non-scaled}}$
K_{m}	-	minor headloss coefficient
β	-	model regression parameter
η_{m}	-	motor efficiency
ψ	-	moving average (MA) order
H_{net}	-	net head
P_{net}	-	net power
H_{f}	-	nodal fixed-grade
f^{NL}	-	non-linear function
$x_{\text{non-scaled}}$	-	non-scaled decision variable
n_{comp}	-	number of components that constitutes the decision variables ($2n_{\mathrm{VSP}} + n_{\mathrm{FSP}} + n_{\mathrm{Valves}}$)
n_{cnodes}	-	number of control/demand nodes
n_{days}	-	number of days
n_{var}	-	number of decision variables
K	-	number of equality constraints
n_{FSP}	-	number of fixed-speed pumps
n_{G}	-	number of generations
n_{hidden}	-	number of hidden nodes
J	-	number of inequality constraints
n_{in}	-	number of input nodes

n_{iter}	-	number of iterations
n_{jet}	-	number of jets
n_{nodes}	-	number of nodes
O_{t}	-	number of observations used for training
O	-	number of observations
n_{out}	-	number of output nodes
n_{pumps}	-	number of pumps
n_{tanks}	-	number of tanks
$n_{P_{\text{e,max}}}$	-	number of times the maximum power is required by the pump
n_{steps}	-	number of time-steps
n_{turb}	-	number of turbines
n_{Valves}	-	number of valves
n_{VSP}	-	number of variable-speed pumps
n_{warn}	-	number of warning messages
y	-	observed variable
$t_{\text{op}}^{\text{FSP}}$	-	operating time for fixed-speed pumps
$t_{\text{op}}^{\text{Valve}}$	-	operating time for valves
$t_{\text{op}}^{\text{VSP}}$	-	operating time for variable-speed pumps
t_{op}	-	operating time
Q_i^{out}	-	outflow in node i (or demand)
η	-	overall (wire-to-water) pump efficiency
φ_{h}	-	partial autocorrelation at lag h
$\mathbf{P}^{\text{best}}$	-	particle best position
η_{P}	-	Pelton turbine efficiency
r_{h}	-	penalty coefficient for equality constraints
r_{g}	-	penalty coefficient for inequality constraints
m	-	period of seasonality (season)
D	-	pipe diameter
L	-	pipe length
h_{L}	-	pipes headloss
n_{P}	-	population size
E_{recov}	-	potentially recoverable energy
λ_{P}	-	power parameter
P_{d}	-	power to be installed in a hydropower scheme
s	-	pressure head exponent
p	-	pressure
T	-	price of energy (or tariff)
η_{prop}	-	propeller turbine efficiency
$\eta_{\text{prop,peak}}$	-	propeller turbine peak efficiency
ω	-	pump angular velocity
r, n	-	pump curve coefficients
η_{p}	-	pump efficiency
h_{p}	-	pump head gain
$h_{\text{p}}^{\text{loss}}$	-	pump headloss
D_{p}	-	pump impeller diameter

M	-	pump relative speed
N	-	pump rotational speed ($N = (\omega/2\pi) \times 60$)
h_0	-	pump shut-off (or cut-off) head
I_{rand}	-	random integer from $1, ..., \nu$
r_1, r_2, r_3	-	random integers from $1, ..., n_{\text{P}}$
R	-	rank number
r_{c}	-	rate of crossover (or crossover probability)
r_{m}	-	rate of mutation (or mutation probability)
$\mathbf{x}_{\text{r}}$	-	reflected point of a simplex
α_{r}	-	reflection coefficient
ν, θ, σ	-	regression parameters
a	-	resistance coefficient
Re	-	Reynolds number
m_{Re}	-	Reynolds ratio exponent
d	-	search direction
ϕ_{s}	-	search step size
Φ	-	seasonal AR order
Υ	-	seasonal differencing order
Ψ	-	seasonal MA order
σ_{s}	-	shrink coefficient
α^{s}	-	smoothing parameter (or discount factor)
t_{step}	-	step size (duration of a time-step)
C_{stop}	-	stop criteria
L_{initial}	-	tank water level in the beginning of the simulation period
L_{final}	-	tank water level in the end of the simulation period
$\mathbf{x}^G$	-	target vector
t	-	time instant
$\varepsilon_{\text{stop}}$	-	tolerance value for the stop criteria
C_{total}	-	total cost (objective function)
η_{transf}	-	transformer efficiency
P_{trans}	-	transition probability
$\mathbf{Z}^{\text{T}}$	-	transpose of the matrix of predictor variables
T^{s}	-	trend component/smoother
$\mathbf{u}^{G+1}$	-	trial vector
Q_{d}	-	turbine design flow
H_{d}	-	turbine design head
η_{turb}	-	turbine efficiency
C_{instal}	-	turbine installation cost
R_{m}	-	turbine manufacture/design coefficient
D_{out}	-	turbine outside runner diameter
Q_{peak}	-	turbine peak efficiency flow
Q_{t}	-	turbine rated flow
H_{t}	-	turbine rated head
D_{t}	-	turbine runner diameter

η_{T}	-	Turgo turbine efficiency
C_{f}	-	unit conversion factor
η_{VFD}	-	variable frequency drive efficiency
aggregDemand	-	vector of demand aggregated steps
y	-	vector of observations over a time period (time series)
$\boldsymbol{\varepsilon}$	-	vector of random errors
$\boldsymbol{\beta}$	-	vector of regression coefficients
aggregTariff	-	vector of tariff aggregated steps
v	-	vector velocity
F	-	weight coefficient (or mutation factor)

Part I

Introduction and opening remarks

1. Water supply systems efficiency - problems versus opportunities

An introduction to the current global situation of the water sector and to the main challenges encountered in water supply systems is presented. The topic concerning the water-related energy consumption is further explored. The concept of smart water grid is introduced as a key for the global improvement of the water sector.

Changes in lifestyles and eating habits together with the world's population growth by roughly 80 million people each year are leading to an increase in freshwater demand of 64 billion cubic meters a year. At the same time, energy demand is also accelerating, with corresponding implications for water prices. It is expected the global water-sector spending to reach the trillions by 2025. By 2050, the pace of urbanisation will be such that achieving universal access to the supply of water and sanitation will cost the developing world an additional 1 % of gross domestic product (US$ 7.6 billion) every year compared to current needs. Consequently, water utilities are facing some main challenges that can be summarised as: population growth, increasing energy costs, water scarcity, climate change, water quality and the design and management of water infrastructure (Grafton, Wyrwoll, White, & Allendes, 2014; SENSUS, 2012; Worldometers, 2015).

"*Water not only feeds bodies, it also feeds countries.*"
(SENSUS, 2012)

The water industry provides drinking water and waste water services to residential, commercial, and industrial sectors of the economy. With a current annual expenditure of US $184 billion (€ 166 billion), including all water and waste water processes, water utilities worldwide are spending around US $40 billion (€ 36 billion) in the production of water from which US $14 billion (€ 12 billion) are being spent on energy for pumping such water (SENSUS, 2012).

Figure 1.1 represents a scheme of the course of water from the source to the final consumers. Although a number of water systems are gravity fed (*i.e.* the storage tanks are located in elevated places), the entire course of water still needs the support of pumping stations (i) to extract the water and transport it to treatment plants, (ii) to transport the clean fresh water to storage tanks, (iii) to distribute the water to secondary tanks and to customers, (iv) to collect the waste water and transport it to treatment plants and, finally, (v) to the disposal (or reuse) of such waste water.

Figure 1.1: Simplified scheme of the main sections of a water supply system from the source to the end users (adapted from ERSAR, 2012).

It is a fact that the water cycle requires large amounts of energy to work, leading to huge operating costs. The impact of pumping in water networks can be demonstrated, for instance, by the following real numbers:

1. In California (most populous US state, 38M inhabitants), 19 % of the electricity consumption in the state is for moving water from the treatment plants to the consumers, corresponding to more than 48000 GWh of water-related energy use per year. More than 14000 water pumps are controlled in the water networks of this state (California Energy Comission, 2005).
2. In Portugal (10M inhabitants), the annual expenses with water supply exceeds the US$ 1784 M (around US$ 740.9 M in bulk water services and more than US$ 1043.7 M in retail services); the pumping energy consumption for water and waste water services is superior to 644 GWh a year. The bulk services (water collection, treatment and transport) account for the control of more than 400 pumping stations while the retail water services (distribution) include the operation of more than 1800 pumping stations (ERSAR, 2012).

The most common current solution for the operation of pumping stations in water networks is typically based on the water levels in the storage tanks. The control operator switches the pumps on to fill the tanks and then switches them off while the consumers are being supplied until those tanks reach again the minimum level. Although this solution works, it is far from being efficient. This type of operation could be highly improved by adjusting the pumps operation to periods of lower electricity price and by using variable-speed drives in order to adapt the pump operation to the water demands and the network requirements. The main challenges associated to these types of solutions are related to the high level of difficulty for a control operator to predict water demand patterns from all type of consumers (from domestic to industrial) by intuition and, at the same time, to deal with the high variability of energy tariffs, with the low capacity of tanks and, moreover, to find adequate control settings for variable-speed pumps without pursuing information about near-future demands. With so complex and large systems, this attempt to improve the operational efficiency could be like a “shot in the dark”, which could lead to serious implications in the minimum requirements for a reliable operation of the networks (such as pressures or security of supply).

Despite some improvements that have been made in the water supply sector, most water networks are still largely inefficient both in terms of operation and management. According to the Black & Veatch’s report (Black & Veatch, 2015), energy efficiency plays a large role in the sustainability effort of water utilities. This is more than expected since, globally, 35 % of the total expenses with water production are being spent on energy. Furthermore, the tendency is the intensification of such concerns with the increases in energy prices.

The main obstacle for the efficiency improvement in water supply systems is mostly related to the complexity of the systems (both in terms of networks configurations and number of variables to

control) and also to the low levels of resilience in the current systems operations. At the same time, factors such as the lack of funding, unfavourable economics and/or lack of political and regulatory support for the implementation of efficiency measures can represent significant challenges to the sector (SENSUS, 2012).

1.1 Smart water grids

Currently, there is no standard definition for the smart water grids. However, the use of innovative (smart) technologies is always attached to it. Smart leakage and pressure management techniques, smart monitoring, smart data management, smart network operations and maintenance and smart techniques for water treatment are topics that usually stand out within this market (ARUP, 2013; SENSUS, 2012). In fact, the integration of the knowledge of information and communication technologies (ICT) in the water networks operation and management is largely contributing for the development of smart water grids, which, in turn, has becoming a driving force for the efficient and sustainable development of the water industry.

SENSUS (2012) enumerates five main components/layers for a comprehensive smart water network solution:

1. Measurement and sensing devices, to collect data from the networks (flow, pressure, quality, etc.);
2. Real-time communication systems, to gather the collected data and/or to send execution actions (*e.g.* pumps or valves shut-off);
3. Data management software, to efficiently handle the collected data by means of automatic processing;
4. Real-time data analytics and modelling software, to obtain useful insights from the network data, monitor and evaluate the potential impact of possible changes in the network (*e.g.* patterns detection, predictive analysis of control scenarios, etc.);
5. Automation and control tools, for the automatic and remote execution of actions/tasks received by means of real-time communication channels.

Figure 1.2 provides an overview of the water course from its extraction to its discharge back into the water course, demonstrating where smart water technologies could be implemented. In such mentioned technologies may be included smart water meters, smart pumps and valves, smart pipes and/or smart monitoring (ARUP, 2013).

It should be noticed that utilities of different dimensions or different economic situations may present distinct adoption rates to this type of technologies and services (SENSUS, 2012). While, on the one hand, a large water utility having a complete monitor and sensing system with communications channels installed (e.g. SCADA) might be willing to implement the next steps for a comprehensive smart water network solution, on the other hand, a small water utility without sufficient IT capacity and data analysis software in-house may not be interested (neither have the capacity) in implementing a full smart solution including all the components previously enumerated. Instead, such utility could prefer, for instance, a "software-as-a-service" (SENSUS, 2012), such as a decision-support software, capable of providing control and management solutions that may be implemented without access to real-time communication systems neither automation or control tools.

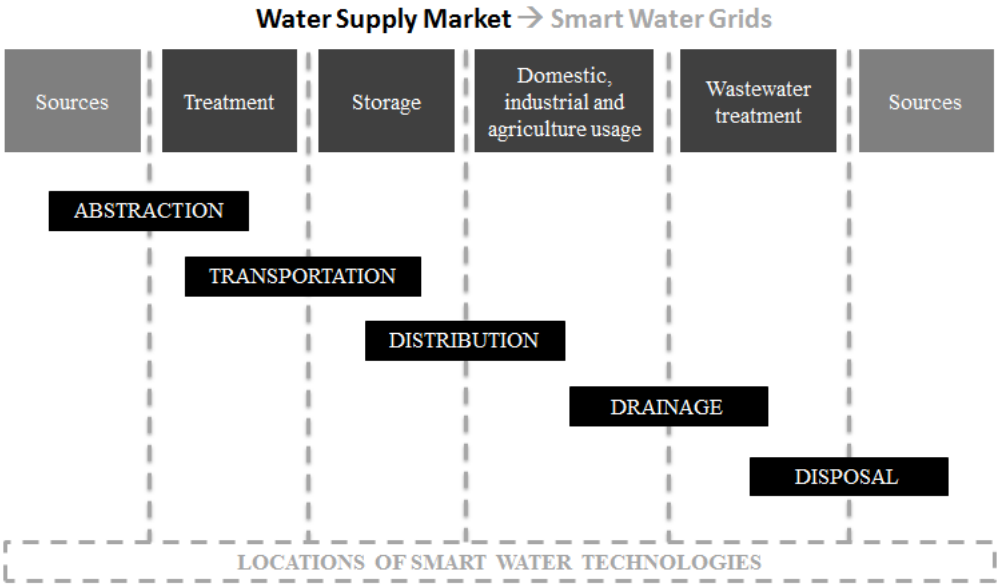


Figure 1.2: Opportunities for smart technologies in the water value chain (adapted from ARUP, 2013).

1.2 Renewable energy sources

The concept of sustainable systems is correlated with reduced carbon footprints. Such reduction can be also achieved with the introduction of renewable energy sources as a response to the huge energy demand in the water and waste water sector, diminishing the consumption of fossil fuels and hence, reducing the carbon emissions. Smart energy grids are already closely tied to technologies for renewable energy generation (ARUP, 2013). It makes all sense to also tie such concepts to the smart water grids. Since the growth in renewable energy is recognised as one of the principal drivers of change in the smart energy market (ARUP, 2013; SENSUS, 2012), a similar approach could also be followed to foster adoption in the water market.

Renewable sources such as solar or wind produce intermittent supply which threaten the stability and reliability of electricity networks. However, the application of such alternative energy sources in water networks may benefit from the storage capacity of this specific networks, *i.e.* it is possible to use the available energy (even if in intermittent periods) to pump the water to storage tanks for a later distribution by gravity.

When the subject is water, the hydropower generation is probably the renewable energy solution that arises in the first place. In fact, a topic that has received particular attention in the last years is related with the energy recovery in water supply systems from the excess of pressure in the networks using turbines (or pumps-as-turbines) for hydropower generation. However, the process of identifying reliable locations for the installation of hydroturbines in the networks is not easy due to the large dimension and complexity of the pipe networks and also the high variability typically associated to the operation of such systems, which also makes this task a very time consuming one.

2. Motivation - a brief market assessment

An assessment of the numerous opportunities in the smart water grids market, both in terms of technology and smart management of water networks, is presented. The main motivational aspects that boosted the work presented in this thesis are described.

Nowadays, it is of the most importance to perform a proper connection between the academic and industrial communities, especially when it comes to engineering fields. The transition from the research & development in the universities to the real application in the industry is only possible when there is a solution to the market needs. For this reason, a brief market assessment that justifies the main motivation of this thesis is presented.

2.1 Smart water grids market

Smart water grid market is a growing industry, especially due to the numerous drivers of change in the water sector. Such drivers are influenced by geography, politics, history, climate and availability of funding. However, the motivation to improve this sector is being essentially driven by a growing public awareness that water is a scarce resource and measures had to be taken to guarantee water supply in an affordable and sustainable manner (ARUP, 2013).

According to SENSUS report (SENSUS, 2012), water utilities could save between $ 7.1 billion and $ 12.5 billion each year from using smart water solutions. These potential savings would come from changes in four key areas in the water sector: (i) leakages and pressure management, (ii) strategic capital expenditure, (iii) water quality monitoring and (iv) network operations and maintenance. The amount of potential savings for each case can be seen in Table 2.1.

Table 2.1: Global opportunities in the water sector - potential savings (SENSUS, 2012).

Amount ($ billion)	Sector
3.4	Leakages and pressure management
4.3	Strategic Capital Expenditure
0.4	Water quality monitoring
1.6	Network operations and Maintenance

In terms of the main goals of water utilities, the improvement of the networks' efficiency and the costs reduction has been highlighted. Figure 2.1 presents the percentage of importance of each key

goal for water utilities. The presented numbers are a result of a survey performed by TaKaDu, Lda., where the participants selected three key challenges related to water network management. As can be depicted, *Improve network operational efficiency* and *Reduce energy consumption* reflects 40 % of the overall goals, making this segment the current largest concern in this industry, followed by the concerns with water losses reduction.

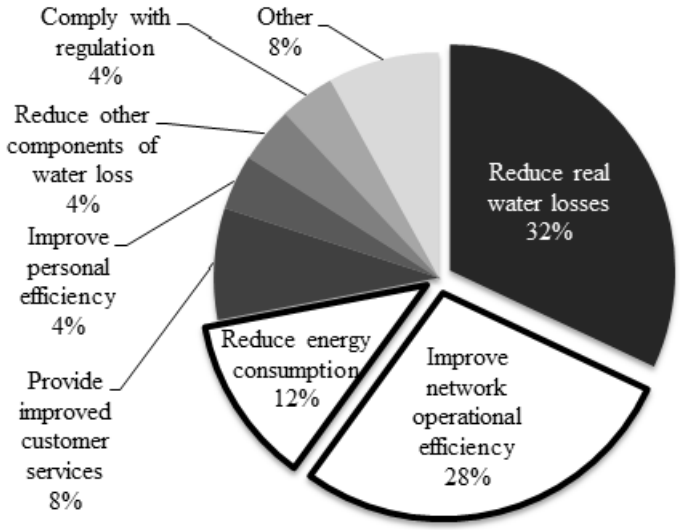


Figure 2.1: Water utilities key goals concerning the water networks management. Results of a survey performed by TaKaDu (adapted from ARUP, 2013).

If, on the one hand, the main savings are identified in the sector of leakages control and pressure management (Table 2.1), on the other hand, the main concerns of water utilities are focused in efficient and low-energy networks operations. Although these seem different goals, there is a strong connection between them. Starting from the first, not only the pressure management contributes to the leakages control but also a reduction in water leakages leads to lower needs of pumping, reducing the energy consumption and improving the efficiency of the operations. At the same time, the improvement of the networks operations is associated to more reliable and stable systems, which may also contribute to stabilise the pressures along the network, and hence, contribute to the losses reduction.

In previous years, a number of measures for efficiency improvement in water supply systems have been applied in several countries (Coelho & Andrade-Campos, 2014). However, most water networks are still inefficiently operated, leading to unnecessary operating costs which reveal a huge market opportunity. The market solutions for achieving efficient operations are essentially related with data analytics and modelling software. Real-time data analysis, demand prediction and modelling represent key activities for achieving optimised operational solutions in a reliable way.

2.2 Market segments and addressable market

The total market for smart solutions for water utilities is a growing market, with an estimated value over $ 5 billion and expected to grow at 20 % per year, reaching over $ 20 billion by 2020. This evaluation includes the total market of smart utilities for investments in innovation, design consultancy, hardware development and installation, automation and control and software (ARUP, 2013). Figure

2.2 depicts the dimension of the smart water market in terms of distinct segments. Although the *smart water infrastructure* represented the largest market size in 2010, the second largest value was for the *ICT, software and analytics* segment, which present the highest annual growth rate (CAGR*) from all the analysed segments.

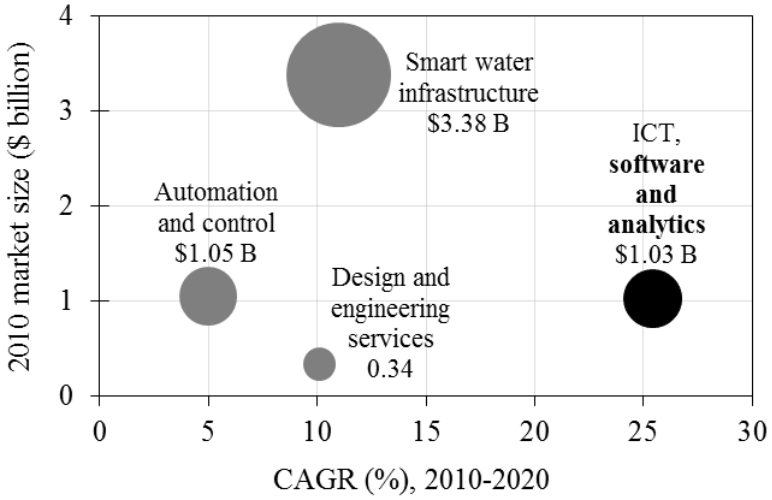


Figure 2.2: Smart Water Grid Market: global vertical market attractiveness by segment, 2010-2020 (adapted from ARUP, 2013).

Figure 2.3 reveals expected returns between 15 % and 20 % for Asia and the emerging markets in Latin America and Africa, respectively. However, the expected strongest markets for smart water grids are located in North America and Europe.

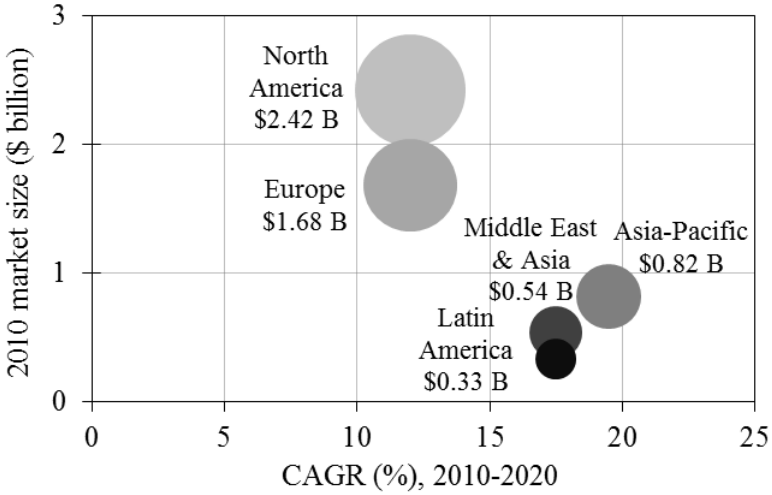


Figure 2.3: Smart Water Grid Market: Regional Attractiveness by region, 2010-2020 (adapted from ARUP, 2013).

For the estimation of the addressable market for data analytics and modelling software, it was considered only two thirds (2/3) of the *ICT, software and analytics* segment, which leads to a total

*Compound annual growth rate (CAGR) between 2010 and 2020 is a business term for a geometric progression ratio that provides a constant rate of return over the time period. It is particularly useful to compare growth rates from different data sets such as revenue growth of companies in the same industry.

addressable market superior to $ 650 million. Considering that this segment is equally distributed between regions (which does not represent accurately the reality, but an approximation) then, considering the total market for smart water grids distributed by region (Figure 2.3), the expected addressable market for data analytics and modelling software, geographically distributed, is represented by the values provided in Figure 2.4.

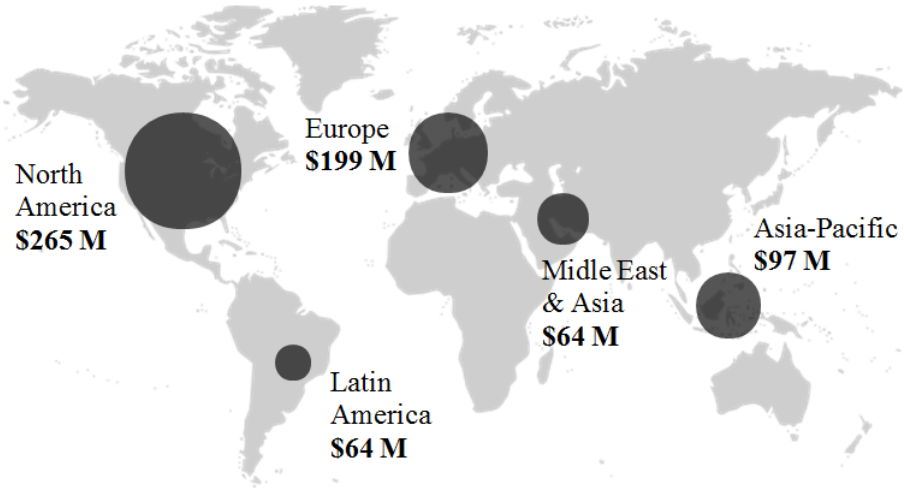


Figure 2.4: Addressable data analytics and modelling software market within the smart water market by region.

Based on the previous criteria, the geographical regions more suggestible to implement software-related products are the markets of Europe and North America, representing over 70 % of the total addressable market. However, it should be noticed that the emerging countries from Asia-Pacific, Latin America, Middle East and Asia may also represent, in the future, attractive markets. Since these countries are starting designing and building new water networks, it may be easier to provide an integrated solution for a comprehensive implementation of a smart water grid.

"*The idea of intelligent systems and smart water systems is here to stay.*"
(Fred Ellermeier, Black & Veatch)

"*All of our findings on smart water networks point to a massive opportunity for utilities and could truly revolutionise water distribution networks around the world - many of which have remained largely static and untouched for decades.*"
(SENSUS, 2012)

3. Thesis objectives and guidelines

The objectives defined for this thesis that resulted from the motivational aspects discussed in the previous chapters are described and justified accordingly. The layout of the thesis is explained, providing guidelines to the reader. An overview on the main contents of the chapters that compose each part of the thesis is provided.

From the previous chapter, it became clear that software-related products have a strong position in the market of smart water networks. At the same time, the research works performed in universities can provide a very important contribution for the development of such kind of solution. Despite that, SENSUS (2012) reported an important concern shared by water utilities that is related with the "*lack of a clear, user-friendly integrated technology solution*".

This thesis intends to provide a first step for a response to such concern. The idea is to develop a computational tool capable of automatically provide optimal solutions for the efficient and sustainable operational control of any water supply network. Two main approaches are followed for achieving the efficiency of the networks operations: (i) application of optimisation techniques for the minimisation of pumping energy costs and (ii) implementation of an automatic methodology for detecting potential locations for turbines installation in the networks in order to reduce energy wastes.

In order to achieve the main objective with success, a number of research objectives must be attained: (1) Review the most important optimisation techniques applied to water supply systems; (2) review the existent solutions for modelling water networks; (3) review the works related with implementation of turbines in water networks for energy recovery; (4) investigate the use of distinct forecasting techniques for predicting short-term water demands; (5) define a methodology for the control optimisation of different water networks; (6) implement the defined optimisation methodology in order to use distinct optimisation algorithms and also to allow further implementations of additional algorithms; (7) test the implemented optimisation methodology with the distinct optimisation algorithms in benchmark water networks characterised by distinct dimensions and distinct number and type of elements to control; (8) define a methodology for the identification of sites in water networks with potential for energy recovery using turbines; (9) implement the defined methodology for energy recovery and turbines selection/design; (10) test the implemented methodology using a comparative case study; (11) disseminate the results obtained from the developed and implemented methodologies incorporated in the computational tool through conferences presentations and papers publication.

As mentioned in the beginning of the chapter, this thesis intends to provide an initial contribution for the development of a completely integrated smart solution. Despite the mission has been the ap-

plication of optimisation techniques and an automatic method for finding locations in the networks for turbines installation, the presented tool was developed with the aim of facilitate future developments including the implementation of additional techniques in order to provide responses to other challenges faced in water supply systems, such as pressure management, leakages control and reduction, predictive maintenance, etc.

The presented document is divided into six parts: (i) Introduction and opening remarks, (ii) Literature review and state-of-the-art, (iii) Methodology and mathematical modelling, (iv) Implementation, (v) Validation, results and discussion, and finally, (vi) Conclusion and closing remarks. Each part is composed of chapters, as described below, organised into sections and subsections. In the end of each part, a list of the corresponding references is provided. Some supporting information is included in the appendices at the end of the document.

Part I - Introduction and opening remarks

This first part of the thesis introduces the main problems and opportunities in current water supply systems, focusing essentially in the challenges for the efficiency improvement (Chapter 1). A brief assessment of the potential of smart water solutions is provided in Chapter 2 in order to demonstrate the main motivation for the development of research work in this thematic. The thesis objectives and layout are described in Chapter 3.

Part II - Literature review and state-of-the-art

The second part of the thesis is composed of three chapters. The first chapter (Chapter 4) provides an introduction to the water supply systems, their current situation and main configurations and explores measures for achieving energy efficiency in such systems, introducing the use of renewable energy sources and mainly the use of turbines for energy recovery. This chapter is also devoted to the state-of-the-art of the hydraulic simulation, exploring the existing computer programmes. In Chapter 5, a review on the application of optimisation techniques in WSS is provided, discussing the main works in the area. Works related with water demand forecasting are also discussed. Finally, Chapter 6 crosses the topics addressed in this thesis with the current industry requirements and provides a general discussion as well as some conclusions.

Part III - Methodology and mathematical modelling

This part of the thesis, comprised of four main chapters, is devoted to the explanation of the main methodologies commonly applied within the topics covered in this thesis (modelling, optimisation, forecasting and energy production), focusing mainly in the mathematical modelling approaches followed in the developed work. The first chapter presented (Chapter 7) is devoted to the methodologies related to the water supply systems modelling; Chapter 8 addresses the proposed optimisation approach and explains the selected optimisation algorithms and constraint-handling techniques for this work; Chapter 9 is devoted to the main methodologies commonly used for the development of time series forecasting models which can be applied for the prediction of water demands; finally, Chapter 10 exposes the methodologies followed for the identification of possible locations for turbines installation in water networks as well as for the selection of the most adequate type of turbine for the identified location(s).

Part IV - Implementation

This part of the thesis is devoted to the implemented methodologies including the detailed description on how such implementations were performed. A single-chapter organised into three sections is presented (Chapter 11). A general overview of the developed integrated tool is initially provided. Then, in each section, the distinct modules that compose the developed tool are described with more detail. The first section goes into detail on the implementation of the simulation module based on the hydraulic simulator EPANET 2.0. The second section describes all the particularities of the optimisation module and the third and last section is devoted to the implementation of the energy recovery module using hydroturbines.

Part V - Validation and results

This part of the thesis provides selected results of the developed work. Four main chapters are presented, each one devoted to particular studies and/or analysis concerning the main topics addressed in this thesis (modelling, optimisation, forecasting and energy recovery). The first chapter (Chapter 12) provides the results concerning the modelling of a simple network under distinct operational and design conditions, including the use of variable-speed pumps. Chapter 13 provides the results obtained from the use of the developed optimisation tool in distinct examples of water networks applying the studied optimisation techniques. In Chapter 14, the forecasting results of the studied models applied to data sets obtained from a Portuguese water network are presented. Finally, Chapter 15 provides the results concerning the application of the energy recovery module.

Part VI - Conclusion and closing remarks

The last part of the thesis is devoted to the conclusions concerning each topic addressed in this work (Chapter 16). As closing remarks, an overview of the main scientific and industrial contributions of the presented work is provided as well as some recommendations for future works (Chapter 17).

Appendices

This part of the thesis is composed of nine appendices. Appendix A provides an overview about design and operational optimisation methods applied to water supply and distribution systems. Appendix B also provides an overview but related with methods for the prediction of water demands. In appendix C, information concerning the roughness coefficient for the computation of headlosses in pipes is provided. Appendix D provides the EPANET input files of the water networks used in this thesis for the validation and testing of the developed and implemented methodologies. Finally, appendix E presents the results for the performance of the ANN-based forecasting models implemented.

References

ARUP. (2013). *BIS research paper No. 136. The smart city market: Opportunities for the UK.* Retrieved 2015, from https://www.gov.uk/government/

Black & Veatch. (2015). *2015 Strategic directions: U.S. water industry report.* Retrieved 2015, from http://bv.com/reports/2015/water/

California Energy Comission. (2005). *California's Water - Energy Relationship* (Final staff report). Retrieved 2015, from http://www.energy.ca.gov/2005publications/CEC-700-2005-011/

Coelho, B., & Andrade-Campos, A. (2014). Efficiency achievement in water supply systems - a review. *Renewable and Sustainable Energy Reviews*, *30*, 59–84.

ERSAR. (2012). General characterisation of the sector. In *Rasarp 2012 - annual report of water and wastewater services in portugal* (Vol. 1). Retrieved 2014, from http://www.ersar.pt/website/

Grafton, R. Q., Wyrwoll, P., White, C., & Allendes, D. (2014). *Global water: Issues and insights*. ANU Press, The Australian National University.

SENSUS. (2012). *Water 20/20: Bringing smart water networks into focus* (White Paper). Retrieved 2015, from http://sensus.webdamdb.com/

Worldometers. (2015). *Water consumption statistics.* Retrieved 2015, from http://www.worldometers.info/water/

Part II

Literature review and state-of-the-art

4. Introduction

An introduction to water supply and distribution systems is provided and distinct measures for improving their efficiency are discussed. Renewable energy sources, including solar, wind and hydropower generation, are presented as potential solutions for the high energy consumption in such systems. The existing available software for modelling water supply and distribution systems are also presented and briefly described.

Water and Energy are essential elements for the well-being of the societies. The world energy consumption for water distribution is about 7 % of global energy (James, Godlove, & Campbell, 2002).

Nowadays, it is observed an increase distance between populations and water sources due to the population growth, leading to fast expansions of several water networks. At the same time, the global water consumption has quadrupled in the last 50 years and it is expected that this value continue to increase (Umweltbundesamt, 2010). Consequently, the immediate consumers supply without any planned strategy has led to inefficient operated systems, increasing the energy costs for water supply and distribution.

With the actual concerns about sustainable development, the improvement of energy efficiency in Water Supply Systems (WSS) must be of major importance.

The improvements of energy efficiency in WSS can pass through simple monitoring operations for leakages control to more complex operations such as the consumption predictions, pump systems optimisation, storage/production reservoir systems optimisation and real-time operations. Computational modelling becomes an important auxiliary tool for these more complex studies of energy efficiency in WSS (Martins et al., 2006).

4.1 Revisiting water supply systems concepts

Water supply and distribution systems should satisfy the requirements of several consumption sectors, responding to the demand in each place, in each time and with appropriate pressures (Viessman, Hammer, Perez, & Chadik, 2009). Despite the large size variation and complexity of water distribution systems, all these have the function of deliver water from the source to the consumer (Walski, Chase, & Savic, 2001).

Generally, a WSS comprises four main sections (Swamee & Sharma, 2008): (i) water sources and intake works, where the water extraction is made by intake structures and pumping stations; (ii) treatment works and storage, (iii) transmission mains (pumping and/or gravity), where the bulk water is

transported to treatment plants and then to storage reservoirs; and (iv) the distribution network which delivers water to consumers through service connections. The distribution networks configuration can be looped, branched or, as in most of cases, mixed (a combination of looped and branched networks). Branched networks, commonly used for rural and industrial water supply (Amit & Ramachandran, 2009), only enable one flow direction whereas looped networks, with connected pipes that constitute loops, allow changes in flow direction according to the different demands in each node. The main advantage of the looped configuration is the guarantee of water supply when some pipe break occurs (for maintenance, for example). Another advantage of looped networks is related with lower velocities (due to the existence of more than one path for water) that enlarge the system capacity (Walski et al., 2001). For these reasons, looped systems are generally more desirable in urban environments (Amit & Ramachandran, 2009).

The main components of a water supply system are (Viessman et al., 2009): pipes, junctions, storage reservoirs (tanks or variable level reservoirs), water sources (or fixed level reservoirs), pumps and valves. In energy efficiency studies of WSS, some concepts about these main components must be taken in account.

Pipes are responsible for carrying water. Headlosses (or energy losses) that occur along the pipe walls are usually called friction losses and can be represented by an expression related with the pipe resistance (or roughness) coefficient (Walski et al., 2001). Headlosses can also occur at other sections of the networks such as valves, bends, reducers, etc., due to turbulence within the flow through fittings and bends. These kinds of losses are usually called minor headlosses (or local losses).

Pumps are essential components on energy efficiency studies of WSS. A pump is a device that transfers the mechanical energy to the fluid as hydraulic head. This head, called pump head, is generally a function of the flow that passes through the pump. The pumps are used when the WSS needs energy to overcome elevation differences and headlosses. Centrifugal pumps are the mostly used in this kind of system (Walski et al., 2001). The relationship between pump head and pump flow rate is represented by the pump head characteristic curve. This is a non-linear curve that shows a decreasing head with the flow rate through the pump. Pumps can be of constant- or variable-speed and must operate inside the limits imposed by the characteristic head curves.

Another important issue concerning the pumps is the operating point given by the crossing point between the pump head curve and the water system resistance curve (see Figure 4.1). The operating point represents the discharge that will pass through the pump and the head that will be added by the pump (Walski et al., 2001). When the system head variation or the water demand variation occurs, the pump can operate outside the nominal work point with lower efficiency conditions (Gomes et al., 2007).

In variable-speed pumps, the pump discharge is directly proportional to pump speed and the pump head is proportional to the square of the speed (Walski et al., 2001). According to this relation, the characteristic curve of this kind of pump shifts as the speed changes (Rossman, 2000).

The valves are elements that can be open or closed to control the movement of the water throughout a pipeline. According to their specific functions, valves are commonly classified into the follow categories (Walski et al., 2001): (i) isolation valves (gate and butterfly valves), the mostly used in WSS, which can be manually closed to block the flow (useful for maintenance and emergencies); (ii) directional valves, also called check valves, which prevent the change of the flow direction throughout the pipeline; (iii) altitude valves, used to control storage reservoirs levels; (iv) air release and vacuum

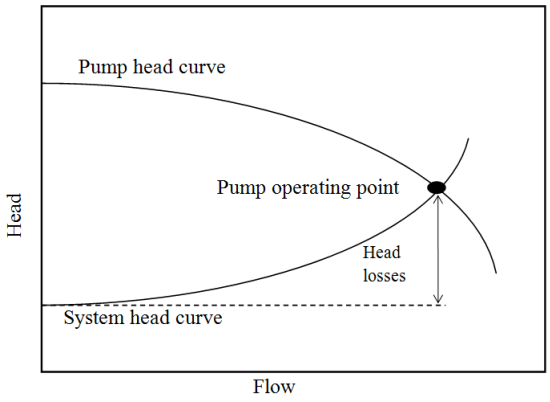


Figure 4.1: Definition of the pump operating point.

breaking valves to release the trapped air in the system or to discharge air upon the system to solve negative pressures; and (v) control valves (or regulating valves), which can be subdivided according to the type of control - Flow Control Valves (FCVs), Pressure Reducing Valves (PRVs), Pressure Sustaining Valves (PSVs) and Throttle Control Valves (TCVs).

The storage reservoirs are also important elements in WSS. Besides being the elements with the higher visibility, they present several purposes (Eleotero, 2008): flow regulation, pressure regulation, supply security and water reserve for fires.

Pumped systems can present distinct configurations that can be characterised from direct pumping from a reservoir to the distribution network to pumping from and to storage tanks available in the network. Walski et al. (2001) enumerates the main existent alternative configurations: (i) pump feeding directly the network (constant or variable speed), (ii) pump feeding through a pressure reducing valve (PRV), (iii) pump with a parallel pressure relief valve, (iv) pump into an hydropneumatic tank, (v) pump feeding directly a network containing a storage tank floating on the system, and (vi) pumped storage configuration (not floating on the system).

4.2 Achieving energy efficiency

The estimative of water loss in the world is around 30 %, meaning that a similar portion of energy is also lost (Feldman, 2009). Multiple factors contribute to these energy losses in water sector (Feldman, 2009): inefficient pump stations, poor design of the networks, installations and maintenance, old pipes with headloss, bottlenecks in the networks, excessive pressures and inefficient operation strategies.

According to Feldman (2009), the main improvements in energy efficiency can be obtained with: (i) pump stations design improvement, (ii) systems design improvement, (iii) variable speed drives (VSD) installation, (iv) efficient operation of pumps and (v) leakages reduction through pressure modulation. The last two topics may present a certain conflict. An efficient pump operation leads to a major use of the pumps during the night (off-peak period) increasing the pressure during that time. On the other hand, the goal of pressure modulation is to minimise pressure over the night in order to reduce leakages. A solution for this conflict can be the isolation of distribution areas (using valves, for example) by means of pressure controlled District Metered Areas (DMA) (Feldman, 2009). This enables the combination of the two methods.

Inefficient pump stations can be caused by inefficient control of the pumps or even by the oversize of the systems. Most existent pump systems are oversized and many of them by more than 20 % (Europump and Hydraulic Institute, 2004), representing an opportunity for the energy efficiency in WSS.

For the flow control in pump stations, bypass lines, throttling valves or pump speed adjustments can be used. However, the pump speed control is the most efficient of these methods (Europump and Hydraulic Institute, 2004). Variable speed drives (VSD) for centrifugal pumps allow their operation with fixed pressure and variable flow or with fixed flow and variable pressure. This allows the reduction of the number of switches (on/off) by the pumps and the reduction of pipe breaks (Feldman, 2009). Furthermore, according to Gellings (2009), these drives have potential to save from 10 to 20 % of total pumping energy and Kiselychnyk, Bodson, and Werner (2009a) indicates a possible energy reduction of 27 % only with 10 % of pump speed decrease.

The difference between the use of control valves or VSD to control the pumps is that, in the first case, with the closure of the valve, an increase on the head occurs, meaning a large dissipation of energy. In the second case, the reduction of the pump operating speed requires a small operating head, implying less power consumption. Fetyan, Younes, Helal, and Hallouda (2007) presented in their work a comparative study between a pump connected to a control valve and a pump using two different kinds of VSD for the speed control (scalar and direct torque control). In both cases, a reduction in the input power with the speed reduction was observed. However, this reduction was larger with the use of VSD applying the direct torque control.

In the cases with no flow rate variation, the use of VSD is not the best choice for saving energy costs (Europump and Hydraulic Institute, 2004). Alternatively, the pump resize, the reduction of the impeller diameters or even the pump replacement for a new one, can be more efficient interventions (Europump and Hydraulic Institute, 2004).

Other measures to enhance the efficiency of the WSS, not referred previously, can be applied, such as (i) the replacement of inefficient equipment, (ii) the leakages management by regular monitoring and maintenance, preventing from both water and energy wastes, (iii) the simple selection of a suitable energy tariff system, or even (iv) the incorporation of renewable energy sources in the systems, reducing fossil fuel dependency.

From all the measures presented, some of them can imply high investment costs, such as the equipment replacements or even the incorporation of new equipment. However, some of the measures do not present significant investment costs when compared with the benefits obtained. Moreover, some measures related to management do not need any significant investment, meaning that, in some cases, the efficiency improvement of the WSS can be obtained without too much effort.

The replacement of some inefficient equipment by high-efficiency pump/motor systems can provide around 10-30 % of pumping energy savings (Gellings, 2009).

In addition to the energy tariff system modification and the introduction of VSD in pumps, Tsutiya (1997) refers some ways to minimise energy costs by reducing the total head of the systems. This can be obtained by reducing the geometric head and the headlosses. Headlosses reductions can be made by (Tsutiya, 1997): (i) the correct choice of the pipe diameters in order to obtain an economic velocity of the water (lower velocities reduce headlosses) or (ii) by the pipes cleaning and/or coating, reducing the roughness of the pipes. Tsutiya also provides the relation between the Hazen-Williams coefficient and the energy cost variations also listed in Table 4.1 (note that the roughness coefficient increases

with the reduction of the H-W coefficient). It should be noticed that, when the Hazen-Williams coefficient pass from 130 to 90, the increase on annual energy costs is 97 % (Tsutiya, 1997).

Table 4.1: Values for the annual energy cost increase when the Hazen-Williams coefficient (or C-factor) is reduced from the value 130 (Tsutiya, 1997).

Hazen-Williams Coefficient	120	110	100	90	80	70	60	50
Energy Cost Increase (%)	16	36	62	97	145	214	318	486

In Tsutiya work it is also shown that the modification of the tariff system can reduce the monthly cost in 50 % and the introduction of VSD on constant speed pumps for flow control can bring energy savings of 38 %. This method has demonstrated to be better than the control by valves using a by-pass system which presented more 42 % of energy consumption (Tsutiya, 1997).

4.2.1 Renewable energy sources

Despite the huge contribute of the previously mentioned measures for the improvement of the energy efficiency in WSS, the dependence of these systems on fossil fuel will still being notorious. The best way to make these systems energetically sustainable is through the introduction of renewable energy sources or even extracting the excess of available energy using turbines, for example.

In effect, due to a large number of advantages (environmental and economic), the implementation of renewable energy production in WSS is becoming very common, increasing significantly the number of studies in this subject.

As shown below, there are essentially three distinct kinds of solution for energy production in WSS, *i.e.*, solutions provided by (1) solar, (2) wind and (3) hydropower generation. The main characteristics of the three types of system actually investigated are briefly described in this section.

Still regarding to these solutions, Chapter 5 addresses a number of works demonstrating successful real applications of renewable energy production in some WSS (essentially wind/hydropower plants), integrating the operational optimisation of the entire systems.

The main obstacle to the implementation of this type of solution for WSS efficiency is related to the implementation cost. In the particular case of hydropower generation, a more economical alternative can be the use of pumps as turbines (PATs) instead of the installation of a hydroelectric plant which requires costs for both land and new equipment.

hydropower generation

Among several alternatives, the installation of micro hydroelectric plants has standing out. In hydropower systems it is usual the use of turbines or pumps operating as turbines (PATs) for the recovery of the excess of energy that is generally lost in the WSS due to the use of pressure reducing valves (PRVs).

As the pumps available in the market are more adequate for reduced power and flows and, at the same time, represent lower investment costs, they present advantages for using in micro-hydroelectric plants (5 to 100 kW) either as pump or as turbine (Gonçalves & Ramos, 2008). Furthermore, according to Ramos, Mello, and De (2010), it is possible to use pumps as turbines with relatively higher

efficiency (up to 85 %). The main disadvantage is the high PAT dependency on flow rate, which do not allows medium and high variations of flow (Caxaria, Sousa, & Ramos, 2011).

Most studies in this field are related to the analysis of the feasibility of the PATs compared to other solutions (Carravetta, Del Giudice, Fecarotta, & Ramos, 2012; Caxaria et al., 2011; Fontana, Giugni, & Portolano, 2011; Lopes & MARTINEZ, 2006).

The study of Lopes and MARTINEZ (2006) demonstrated that the use of pumps as turbines (PATs) can represent a viable choice especially for installations inferior to 4 kW. A study-case provided feasible options with times of investment return from 4 to 22 months maximum (Lopes & MARTINEZ, 2006).

Caxaria et al. (2011) compare the performance of the use of a PAT with the use of a five-blade propeller turbine for hydropower generation. The five-blade propeller turbine demonstrated to be a very promising solution, with high hydromechanical efficiency values. Although the application of a PAT is a viable choice, the five blade turbine has the advantage of non-interference with the normal flow behaviour in the piping.

The work of Fontana et al. (2011) provides a study of the use of PATs instead of PRVs for losses reduction and energy production in WSS. Experimental tests were performed in the city of Naples (Italy). At an initial phase, the authors used a simulation model and genetic algorithms for the optimal location of PRVs for losses reduction. Then, the global or partial replacement of the PRVs by PATs was implemented for hydropower generation. Although the optimal location of PRVs for losses reduction does not maximise the energy production, results have shown that a relatively large energy recovery can be obtained with significant reduction in water losses.

Carravetta et al. (2012) proposed a PAT design method, based on a variable operating strategy, for the identification of the PAT performance curve that maximises the produced energy for a certain flow and pressure head distribution pattern. The authors pointed out two main problems related to the design of a small hydropower plant: (i) the lack of complete series of characteristic curves of industrial PATs and (ii) the need of a strategy for turbine selection.

Photovoltaic and wind power generation

Wind water pumping, resorting to mechanically coupled wind turbines, has been used since ancient history. However, more recently, turbines have also been electrically coupled (Muljadi, 1997). The advantage of the electrical coupling is the location of the wind turbine that is independent of the water pumping location (Muljadi, 1997). However, photovoltaic water pumping systems are actually being applied especially in WSS with poor electrical requirements (Kiselychnyk et al., 2009a).

The implementation of optimisation strategies for the operational improvement of water supply systems containing solar- and/or wind-hydropower plants requires the availability of power forecasts, which can be possible to obtain, for example, through time-series analysis or prediction algorithms based on neural networks or fuzzy logic (Castronuovo & Lopes, 2004).

The work of Muljadi (1997) provides an analysis of the dynamics of a wind-turbine water pumping system. The analysis process was illustrated by the simulation results of the system. It was observed that the operating point of the wind turbine was affected by the motor and the water pump characteristics. Non corresponding wind turbine characteristics with water pump characteristics (such as the size) lead to an efficiency degradation and also to a reduction in the operating rotor speed range.

Kolhe, Joshi, and Kothari (2004); Muljadi (1997); Nayar, Vasu, and Phillips (1993); Vongmanee (2005) provided works dealing with reasonable efficient photovoltaic (PV) water pumping systems.

The overall optimal operation of a PV pumping water system is only achieved if the transformed mechanical load, converted by the electric motor, matches the maximum power line of the PV generator and if that power line matches the maximum hydraulic output of the pump (Nayar et al., 1993). Additionally, Kolhe et al. (2004) tested the effect of changing the orientation of the PV array and concluded that the output obtained is 20 % superior to the compared fixed PV array.

4.3 Hydraulic simulation

Simulation models are computational representations and/or reproductions of real systems behaviour through functions (Walski et al., 2001). Hydraulic simulators are numerical programmes in which it is possible to implement models for water transportation and distribution. These models replicate the non-linear dynamics of the networks by solving a set of hydraulic equations including conservation of mass and conservation of energy (Machell, Mounce, & Boxall, 2010). Therefore, this kind of tool allows the users to get details about all elements of a certain network represented at specific times, providing an important support for management and operational control even for the most complex systems. However, a unique approach for modelling does not exist, not even for the simplest WSS.

The first pipe network digital models appeared with the coming of the digital computers and the FORTRAN programming language between the sixties and the seventies (Walski et al., 2001). Even during the seventies, models became more powerful by running not only steady-state simulations but also extended period simulations and later, in early eighties, the first water quality model was developed (Walski et al., 2001).

4.3.1 Types of hydraulic models

Diverse types of models based in hydraulic equations have been used, such as mass-balance models, regression models, simplified hydraulics and full hydraulic simulation models (López-Ibáñez, 2009). More recently, the recourse to Artificial Neural Networks for capturing the knowledge base of a hydraulic simulator, reducing the computational burden, is being proposed and applied (see, for instance, Rao and Salomons (2007)).

A mass-balance model is the simplest method of calculation. This model considers only the flow rate variations in a tank assuming that a pump or some pumps generates the levels variations in the tank. However, the pump's head and the minimum pressure at nodes are neglected (López-Ibáñez, 2009).

Regression models are more accurate than mass-balance models. This kind of model is based on a set of non-linear equations obtained with the responses of a certain network subject to distinct demands. The problem of this model is the sensitivity to the data used for the construction of the model, meaning that some changes in the network can produce invalid results (López-Ibáñez, 2009).

The method of simplified hydraulics incorporates the effect of connected components that compose the network into a single equation. In particular cases, some linear equations are enough to represent the system hydraulics (López-Ibáñez, 2009). However, it becomes different in complex real cases. Models based in a full hydraulic simulation are robust in terms of system modifications and

demand variations (López-Ibáñez, 2009). These kinds of models solve both the equations of mass and energy conservation and these are the most used currently.

Over recent years, there has been a significant increase in the number of software applications in this field (Schmid, 2002). The appearance of such hydraulic simulators has developed from trial and error to more advanced optimisation methodologies (Machell et al., 2010) in order to improve the efficiency of the WSS.

4.3.2 GIS integration

Nowadays, the integration of Geographic Information Systems (GIS) with hydraulic simulators is quite common (*Dorsch Gruppe: DC Water Design Extension*, 2006; Engineering, 2008; *Hunter GIS: EPANET INP Support Module*, 2012; Macke, 2012; REDHISP Group, 2004; Zonum Solutions, 2008a). A GIS is a system that allows capturing, managing, analysing and displaying information geographically referenced (Geographic Information Systems, 2012). These systems are useful for the management of projects involving large volume of data and for the application of some analytical tools. In hydraulic simulation works, importing the model results into GIS provides high quality result display and additional analysis possibilities (Macke, 2012). Therefore, this tool can be used as a source for modelling data and for decision support (Walski et al., 2003), aiding with time and cost savings and contributing to the efficiency improvement.

The typical benefits of combining GIS with a hydraulic simulator are (Macke, 2012): (i) automatic calculation of the pipes length, (ii) a map display with more details, scale-dependent, more flexible, etc., (iii) advanced editing capabilities, (iv) interpolation of the elevation data and (v) demand calculation. Contrary to some hydraulic simulators, in GIS, pumps and valves are usually represented by points (or nodes) and not by links, which requires a special treatment (Macke, 2012).

4.3.3 Existing software

Currently, several software programmes for the hydraulic simulation incorporates a number of others additional tools including SCADA systems, optimisation and calibration modules. Some of these programmes are available in free versions and most of them have no limitation in the networks size.

EPANET 2.0, for example, is a free open source software, developed by EPA (U.S. Environmental Protection Agency), that performs extended period simulation of hydraulic and quality behaviour within pressurized pipe networks (EPA - Drinking Water Research, 2012). This simulator is characterised by a robust model with a large community of users in the world (Vieira & Ramos, 2009), offering an optional user interface and no limitation on the network elements number. It allows the use of metric or US units and supports the commonly used headloss calculation: Darcy-Weisbach, Hazen-Williams and Chezy-Manning.

Several applications using EPANET have been developed, such as:(1) EPANET Z displays online maps/imagery as a background (Zonum Solutions, 2008b); (2) EpaSens performs sensitivity analyses to the network parameters (Zonum Solutions, 2008c); (3) epa2GIS exports the network map and outputs from EPANET to a Geographic Information System environment (Zonum Solutions, 2008a); (4) GHydraulics also integrates EPANET with GIS and calculates economic diameters for specific flow rates (Macke, 2012); (5) GISRed integrates EPANET with a ArcView GIS (REDHISP Group,

2004); (6) EISM is another add-on that allows to import/export data for MapGuide through INP format (*Hunter GIS: EPANET INP Support Module*, 2012); (7) DC Water Design Extension is another ArcView solution (*Dorsch Gruppe: DC Water Design Extension*, 2006); (8) HydrauliCAD, an AutoCAD hydraulic analysis water modelling programme that uses EPANET calculations (*HydrauliCAD*, 1998); etc. Due to all these facilities and the fact of being of public domain, EPANET is the hydraulic simulator mostly used in academic field. However, several commercial programmes are already applied in industry. A number of these existing commercial computer programmes employ EPANET as a basis for the hydraulic modelling and separate modules for the networks optimisation. This is, for example, the case of:

- AQUIS (*AQUIS: 7-technologies*, 2012), a water distribution modelling and management programme that includes not only hydraulic simulation but also pipe design and control optimisation and integrates a calibration module, SCADA and GIS systems;
- Aquadapt (*Derceto, Inc: Introducing Derceto Aquadapt*, 2011), that allows to obtain the optimal operations (with minimum energy) of an entire network making use of SCADA facilities;
- ENCOMS/CAPCOMS (*Halcrow: Water supply and distribution systems optimization software*, 2011), that incorporates calibration, pipe design and control optimisation modules with a SCADA system;
- Helix delta-Q (*Helix Technologies: Helix delta-Q Pipe Networks*, 2011), that only allows the design optimisation of the networks;
- H2ONET/H2OMAP (Innovyze, 2012) provides calibration, pipe design and control optimisation, GIS and SCADA facilities;
- Mike Net (*DHI: MIKE - MODELLING THE WORLD OF WATER*, 2011), that incorporates optimisation and calibration modules, uses a GA and includes SCADA and GIS facilities;
- optiDesigner (OptiWater, 2012), that also uses GA to find the least-cost design of the networks;
- Optimizer WDS (Optimatics, 2011), that allows design and operational optimisation in real-time and the application of distinct optimisation algorithms for distinct cases, such as Evolutionary Algorithms, Genetic Algorithms, Non-Linear Programming and also Artificial Neural Networks when a reduced computer run-time is needed;
- SynerGEE Water (*GL Water*, 2012), that includes GIS and SCADA system and provides a module for pipe design optimisation;
- STANET (STANET, 2010), like the SynerGEE Water, includes GIS and SCADA facilities and a pipe design optimisation module;
- Wadiso (Doig, 2002), also for the optimal design of the networks, integrating GIS and SCADA systems;
- WaterCAD/WaterGEMS (*Bentley: WaterCAD V8i*, 2010), a robust hydraulic simulator based on EPANET that incorporates GIS facilities (WaterGEMS) and both calibration, optimisation (design and control) and SCADA modules.

Other kind of commercial hydraulic simulators not based on EPANET are also available in the market, such as:

- Aquadapt (*Derceto, Inc: Introducing Derceto Aquadapt*, 2011), that includes management optimisation of the networks making use of GIS and SCADA facilities;
- AquaNet (INAR, 2003), a simple pipe system modelling software;
- Cross (rehm, 2012), another computer programme that only performs pipe systems simulation;
- Eraclito (PROTEO S.p.A., 2012), a hydraulic simulator with similar characteristics to the Aquadapt;
- HYDROFLO (Tahoe Design Software, 2010) also for hydraulic simulation; the developers of this simulator also offers the PumpBase 2.0, an advanced pump selection software;
- MISER (Tynemarch Systems Engineering Ltd, 2012), similarly to Aquadapt and Eraclito, this is a hydraulic simulator that also allows the optimisation of the networks management, incorporating GIS and SCADA systems;
- Pipe2012 (KYPipe, 2009) is a more advanced computer programme for water network management, grouping modules such as calibration and optimisation (design and control) and using GIS and SCADA facilities.

Although EPANET 2.0 is actually the most widely used hydraulic simulator, other public domain computer programmes exists, however, with some limitations. It is the case of:

- Branch/Loop, where Branch calculates the least-cost design of branched water distribution networks using linear programming and Loop simulates the hydraulic behaviour of looped networks. The main disadvantage of Branch/Loop software is the limitation on the size of the networks (Schmid, 2002).
- NeatWork (NeatWork, 2010), that uniquely determines the optimal design of water gravity networks for rural areas.

5. Water supply systems optimisation

The main techniques that have been applied for design and operational optimisation are presented and discussed. Studies concerning networks operational optimisation in real-time and also including the production of renewable energy in their operations are presented. A review on the most applied techniques for water demand forecasting is presented. A specific focus is given to short-term forecasting techniques, namely, hourly, daily and weekly forecasts.

The development of water supply systems without the use of optimisation provides non-optimal structures, based essentially on the immediate response to the growing water demand of population and industry (Kiselychnyk et al., 2009a). These non-optimal structures are translated into non-efficient systems in terms of design and operation.

Although reproducing the hydraulic behaviour of the systems, a hydraulic simulator does not allow the determination of the optimal structures or the optimal operational conditions of the systems. For these reasons, the use of optimisation tools is crucial.

Optimisation problems can be solved using conventional *trial and error* methods or more effective optimisation methods. However, in water supply systems, the optimisation process by *trial and error* methods can present difficulties due to the complexity of these systems: multiple pumps, valves and reservoirs, headlosses, large variations in pressure values, several demand loads, etc. For this reason, innovative non-linear optimisation algorithms are becoming more widely explored in optimisation processes of the water supply systems.

Actually, there is no "perfect" algorithm to solve all the optimisation problems. For the WSS case, this can be also observed. During the past decades, a large variety of non-linear optimisation techniques have been applied for the design and operational optimisation of the water networks.

Non-linear optimisation algorithms can be distinguished according two general classifications (Coelho & Andrade-Campos, 2012): (i) classical algorithms, based essentially on the computation of the objective function gradient and/or function evaluations and (ii) heuristic algorithms, consisting essentially on exploratory search and generally based on phenomena that occur in nature or even based on artificial intelligence.

The classic algorithms applied in WSS optimisation comprise: Linear Programming (LP), Non-linear Programming (NLP), Integer Non-linear Programming and Dynamic Programming. These kinds of algorithms enable finding the exact position of an optimal solution. However, they usually converge to local optimal solutions which could not be the global optimum. In addition, the need of

derivative evaluations can make, in some cases, the optimisation process more complex.

From the group of heuristic algorithms, it is usual to find works applying mostly Genetic Algorithms (GA) and Evolutionary Algorithms (EA). However, other techniques have also been investigated, such as Particle Swarm Optimisation (PSO), Tabu Search (TS), Ant Colony Optimisation (ACO), Simulated Annealing (SA), Shuffled Complex Evolution (SCE) and Harmony Search (HS). These techniques provide the advantages of not requiring derivatives calculations and not relying on the initial decision variables (Coelho & Andrade-Campos, 2012). Due to the exploratory nature of the heuristic algorithms, the probability of finding global optimal solutions using these advanced techniques is higher (Coelho & Andrade-Campos, 2012). On the other hand, the main disadvantage of these methods is related to the higher computational effort (Coelho & Andrade-Campos, 2012).

The concept of hybrid algorithms combining global optimisation with local search techniques in order to increase the convergence is also largely observed in the literature for the optimisation of WSS.

Some researchers frequently explore the optimisation problems through a multi-objective perspective, dealing with the minimisation (or maximisation) of a number of functions or even dealing with conflicting objectives, which imply the minimisation of some functions and, at the same time, the maximisation of other functions.

The goal of a multi-objective problem (MOP) is then to optimise (minimise and/or maximise) a number of objective functions simultaneously (Coello, Lamont, & Van Veldhuizen, 2007). Cheung, Reis, Formiga, Chaudhry, and Ticona (2003) pointed out five objectives that constitute a complex MOP for a WSS: (1) hydraulic capacity, (2) physical integrity, (3) flexibility, (4) water quality and (5) economy. However, in the literature, most MOPs applied to WSS optimisation are represented by two general objectives: costs minimisation and hydraulic benefits maximisation.

Multi-objective optimisation methods have the advantage of providing a set of optimal solutions, called Pareto optimum, instead of a unique optimal solution (Coello et al., 2007). This allows the system operator to analyse the set of Pareto optimal solutions and choose one solution considering additional criteria.

Evolutionary algorithms are usually the most used for solving MOPs. While the evolutionary methods deal with a set of solutions during the search procedure, allowing to obtain a set of Pareto optimal solutions in a single run, the classic methods only lead to a single solution and cannot guarantee the generation of different points on the Pareto front (non-dominated Pareto solutions) (Coello et al., 2007).

Respecting to the constraints to which the objective function should respect in order to not reproduce infeasible solutions, Michalewicz (1995) and then Coello et al. (2007) provided reviews about the state-of-the-art constraint-handling techniques applied on evolutionary computation. Michalewicz (1995) classified the techniques possible to be used in nature-inspired algorithms as: (1) penalty functions, which can be static (function of the degree of violation of constraints) or dynamic (function of both the degree of violation and the number of iterations); (2) rejection of infeasible individuals; (3) specialised operators; (4) the assumption of the superiority of feasible over infeasible solutions; (5) behavioural memory; (6) repair algorithms; (7) multi-objective optimisation; (8) co-evolutionary models or even (9) cultural algorithms. Coello et al. (2007) follow a similar classification and also point out the specific advantages and disadvantages of each type of constraint-handling technique.

Recently, Mallipeddi and Suganthan (2010) also provided a work dealing with constraints. Mo-

tivated by the fact that different constraint-handling techniques can be effective in distinct stages of the optimisation, the authors proposed and tested the behaviour of an ensemble of constraint-handling techniques (ECHT) to solve constrained real-parameter optimisation problems. The ensemble is composed of four distinct techniques. Results showed that the ECHT outperformed each the constraint-handling method that constitutes the ensemble (Mallipeddi & Suganthan, 2010).

5.1 Design optimisation

Design optimisation problems in WSS are based on searching the system characteristics which minimise the total system cost without affecting the proper operation of the hydraulic system and the consumers supply. This means that the system must be economic and reliable. However, reliability increase can imply higher costs (Swamee & Sharma, 2008).

According to Gellings (2009), pipeline optimisation can save from 5 to 20 % of pumping energy.

The main obstacles to design highly-efficient systems are (Amit & Ramachandran, 2009; Kiselychnyk, Bodson, & Werner, 2009b): their complexity, spatial distribution, changeable structure, time-varying parameters, availability of discrete and continuous control actions and large range of possible combinations of pipe materials.

Typically, in this kind of optimisation problem, the objective function is expressed in function of costs that can be associated to distinct water supply components such as sources/pumping plants, pipelines, reservoirs and residential connection or even costs associated to energy consumption and establishment costs related to the land, to the operational staff or other facilities (Swamee & Sharma, 2008).

The total costs can also be classified into two main types (Swamee & Sharma, 2008): (1) capital costs, associated to the initial investment and (2) recurring costs, required to keep the operational conditions. Thus, the general WSS design problem can be formulated as the minimisation of the total costs represented by the sum of these two main types of cost (capital and operational), subject to the conservation laws of mass and energy, to the water demand constraints and to the nodal head requirements (Amit & Ramachandran, 2009). Nevertheless, other constraints can also be considered in order to improve the model, such as constraints related to the layout, multiple loadings, uncertainty due to lack of information, operations, water quality, reliability and rehabilitation (Amit & Ramachandran, 2009).

Swamee and Sharma (2008) proposed single expressions, essentially dependent on the materials and dimensions of the elements, for the contribution of each water supply component (pumps, pipes, high-pressure pipes, service reservoirs, surface reservoirs and service connexions) for the total capital costs.

The annual recurring cost of energy consumed in maintaining the flow (or pumping energy cost) can be obtained by multiplying the average pump power by the electricity cost and the total number of hours in a year.

As referred by Amit and Ramachandran (2009), in order to develop a good model for the design optimisation of WSS, some aspects must be included: (a) pipe layout and sizes, (b) location and capacity of tanks, (c) location, types, capacity and operating schedule of pumps and (d) location, types and settings of valves. Additionally, multiple loading demands, reliability, uncertainty and water quality should also be considered in order to satisfy the requirements of real WSS.

The design optimisation review paper published by Amit and Ramachandran (2009), based on the networks configurations, provides distinct models developed either for branched or for looped networks.

Ostfeld and Tubaltzev (2008) classified the design optimisation models developed and published, since the seventies, into six different types, according to the optimisation methodology applied: (1) Decomposition, where the problem is solved using linear programming for a number of fixed flows and the alteration of the flows is made by gradient-based methods; (2) Simulation and Non-linear Programming, based on a connection between a network simulator and a non-linear algorithm; (3) Non-linear Programming, models based only in the use of a non-linear programming formulation; (4) Evolutionary/Meta-heuristic Methods, where, most of times, the Genetic Algorithms are used, but also the Simulated Annealing, the Ant Colony Optimisation, etc.; (5) Multi-objective Evolutionary Methods, which evaluate the least-cost design in parallel with other related objectives; and (6) Other methods, like Dynamic Programming and Integer Programming.

In most works dealing with optimisation design of WSS, some benchmark networks have been constantly used for the comparison between distinct developed methodologies: (a) the two-loop network, a gravity network with a single source, firstly introduced by Alperovits and Shamir (1977), (b) the two-reservoir network, introduced by Gessler (Gessler (1985) *apud* Simpson, Dandy, and Murphy (1994)), (c) the New York City Tunnels (Schaake et al. (1969) *apud* Dandy, Simpson, and Murphy (1996)), (d) the Hanoi network in Vietnam (Fujiwara and Khang(1990) *apud* Liong and Atiquzzaman (2004)) and (e) the Anytown, USA, introduced by Walski et al. (1987). Both networks are represented in Figure 5.1.

The objective of the simple two-loop problem (see Figure 5.1a) is to modify the pipe diameters in order to find the least cost.

The problem of the two-reservoir network (Figure 5.1b) consists in the diameter selection of five new pipes and the clean, duplication or left alone of three existing pipes. The system also includes three demand patterns to satisfy.

The New York City Tunnels problem (Figure 5.1c) consists essentially of a single source in Hill View and two main city tunnels. The objective of this problem is to determine the need of laid a new pipe paralleling to the existing ones and also to determine their diameters. A unique demand case is considered.

The Hanoi network (Figure 5.1d) contains three loops and also ramifications. The objective of this problem is to find the least-cost diameters for all pipes while respecting the minimum value for the head pressure at each node.

Finally, the Anytown problem (Figure 5.1e) is the more realistic system, providing some typical features and problems such as pump and tank sizing and location and also pipe sizing. However, the problem does not consider multiple pressure zones neither multiple demand loads.

Alperovits and Shamir (1977), Kessler and Shamir (1989) and Eiger, Shamir, and Ben-Tal (1994) are examples of works whose decomposition methodology was applied.

First proposed by Alperovits and Shamir (1977), the decomposition approach consists in a hierarchical two-stage decomposition of the optimisation problem. In the first stage, the flows are provided by the user and the local optimal conditions of the network (pipe segments and nodal pressure heads) are obtained solving a linear problem. For the second stage, the authors propose the calculation of the gradient of the total cost with respect to the changes in flow in order to find the flows which pro-

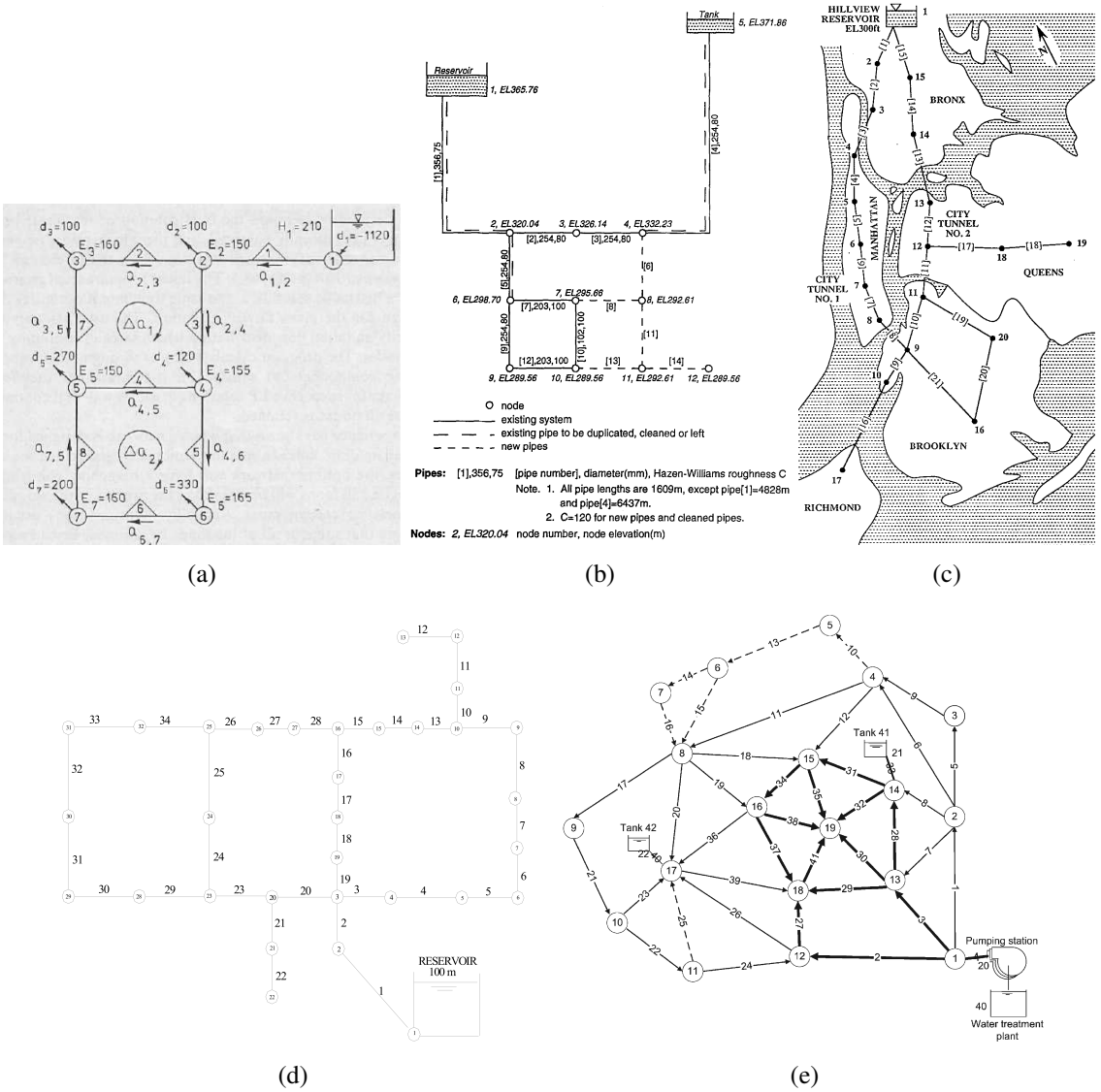


Figure 5.1: Representation of the most tested benchmark water supply networks for design optimisation: (a) two-loop network (Alperovits & Shamir, 1977), (b) two-reservoir network (Gessler (1985) *apud* Simpson et al. (1994)), (c) New York City Tunnels (Schaake *et al.* (1969) *apud* Dandy et al. (1996)), (d) Hanoi (Fujiwara and Khang (1990) *apud* Liong and Atiquzzaman (2004)) and (e) Anytown (Walski *et al.* (1987) *apud* Farmani et al. (2006)).

vide the minimum cost. The main advantages of this linear programming gradient model is that (a) it deals with dimensions, locations, capacities and/or operation of the several elements of the networks, (b) deals with multiple loadings and, (c) for each loading, it provides hydraulically feasible designs (Alperovits & Shamir, 1977). On the other hand, some weaknesses can also be pointed, such as (i) the need for selection of the initial flow distribution of the networks, (ii) the objective function do not reflect aspects like performance and reliability of the networks and (iii) the flows in reservoirs are considered fixed, meaning that their capacity is not considered (Alperovits & Shamir, 1977).

Kessler and Shamir (1989) reformulated the previous model applying matrix notations by mean of graph theory formulation. This change in the formulation has brought independence to the gradient of the objective function of the sets of loops and paths selected in the first stage. The search in the second stage was improved by using the projected gradient method.

The works following the approach of Alperovits and Shamir using gradient calculations ignored the fact that it is not always possible to calculate the gradient. An example to overcome this problem can be seen in the work of Eiger et al. (1994) that presented a strategy applying a global search in the decomposition method considering the gap between the solution and the global optimum. The authors applied a Branch and Bound algorithm that reaches the solution through the combination of a primal process to improve local solutions with a dual process to compress the global bound. It was observed that the algorithm stops in a defined gap between the best value obtained and the lower global bound. When compared with previous results obtained with a decomposition method, for the same examples of networks (including the two-loop and the Hanoi networks), this new approach using global search seems to have greater potential. However, also in this model, there were no considerations to the systems reliability or neither with water quality constraints.

A reliable WSS is a system that satisfies nodal demands and pressure heads for a number of possible pipe failures (Su, Mays, Duan, & Lansey, 1987). To determine the reliability of a WSS, the Minimum Cut-Set Method was considered by Tung (1985) as the most efficient method when compared with other five techniques: (1) Conditional Probability Approach, (2) Tie Set Analysis, (3) Connection Matrix Method, (4) Event Tree Technique and (5) Fault Tree Analysis. A minimum cut-set is a set of components of the WSS that makes the system fail only when failures occur in the entire set (Su et al., 1987). The main failures in water systems occur essentially due to corrosion, excessive load or temperature (Mays, 1989).

Su et al. (1987) introduced continuity and reliability constraints in their model. The developed model is composed by three linkable modules: (1) a steady-state simulation module which incorporates a KYPIPE simulator (referred in section 4.2), (2) a reliability module and (3) an optimisation module based on a generalised reduced-gradient method (Su et al., 1987).

In the work presented by Su et al. (1987), the reliability of a system is expressed in terms of failure probability of the minimum cut-set that can be determined by the Poisson probability distribution. The disadvantages pointed in the model developed by Su et al. (1987) are: (1) the resulting pipe diameters that could not be commercially available, (2) the considerable computational effort required for large looped networks, (3) no multiple loading conditions were considered and (4) the need to incorporate several components (pumps, tanks, valves, etc.), that difficult the problem when the reliability for each component must be defined.

Mays (1989) provides in detail definitions of reliability for each component of a WSS.

Over the years, Genetic Algorithms (GAs) have demonstrated to be effective at solving water

design optimisation problems, however the search for optimal or near optimal solutions become more difficult with the increase of the problems dimension.

Simpson et al. (1994) provided a comparative study between GAs and other techniques for pipe optimisation: (1) Complete Enumeration and (2) Generalised Reduced Gradient (through a non-linear package based on this algorithm). The three methods were used to optimise the two-reservoir network and the results were compared with the firstly obtained by Gessler (Gessler (1985) *apud* Simpson et al. (1994)). Multiple loading conditions were considered and a steady-state solver was used for the hydraulic analysis of the network. Using the Complete Enumeration, the same solution obtained previously by Gessler and also an even better solution were found (Simpson et al., 1994). This technique demonstrated to be effective for networks with few pipes (Simpson et al., 1994). With respect to the non-linear method applied, it looked to work well for small extensions of the networks (Simpson et al., 1994). The GA was particularly effective in finding global optimal or near-optimal solutions with the advantage of providing alternative solutions which sometimes could be preferred (Simpson et al., 1994). In this case, binary strings were used for the codification of the available pipe sizes.

Dandy et al. (1996) presented an improved GA that led to better results than the simple GA. The improvements include (Dandy et al., 1996): (a) use of Gray codes, (b) use of an exponent fitness scaling, where the value of the exponent is adapting and increases in generations without stagnation, and (c) application of an adjacent (or creeping) mutation operator, based on the replacement of a complete decision variable substring by an adjacent possibility from the list of decision variables. The improved GA was applied to the New York City water supply tunnels. The obtained results were compared with solutions using other techniques (Linear Programming, decomposition techniques and heuristic methods) and improvements were achieved with feasible solutions (Dandy et al., 1996).

After discover, in previous works, inconsistencies in network performance predictions caused by distinct interpretations of the Hazen-Williams equation, Savic and Walters (1997) tested the application of different numerical conversion constants C_f for the headloss equation that can be defined by (Walski et al., 2003):

$$h_L = \frac{C_f L}{C^b D^{4.87}} Q^b, \tag{5.1}$$

where L and D are, respectively, the length and the diameter of the pipe, C represents the Hazen-Williams roughness coefficient, Q is the pipe flow rate and $b = 1.85$ (SI units). The use of distinct values for this conversion factor was also observed in later works (see, for instance, results presented on Appendix A.3).

The developed computer model GANET (a GA in cooperation with EPANET) was used and the values tested for the constants were $C_f = 10.9031$ and $C_f = 10.5088$. The benchmark networks tested were (1) the two-loop, (2) the Hanoi and (3) the New York City Tunnels, and distinct results were obtained for each constant used. It should be pointed out that their results were improved compared to some obtained by other researchers, demonstrating the potential of GANET for the design optimisation. The improvements applied to GA in the GANET include (Savic & Walters, 1997): (a) Gray codes for the variables representation instead of the common binary codes and (b) penalty terms added to the fitness function in the case of pressure-infeasible solutions (the penalty is a function of the distance from the feasibility).

Djebedjian, Herrick, and Rayan (2000) tested the two-loop network design optimisation dealing

with constraints by applying the Sequential Unconstrained Minimisation Technique (SUMT) proposed by Fiacco and McCormick (Fiacco & McCormick (1964) *apud* Djebedjian et al. (2000)) however firstly introduced by Carroll (1961) *apud* Djebedjian et al. (2000)). Results were identical to that obtained previously by Savic and Walters (1997).

Wu and Simpson (2001) investigated the use of Genetic-Evolutionary optimisation algorithms in water networks. The authors developed an algorithm, called messy GA, which is characterised by a modified GA according to the following issues (Wu & Simpson, 2001): (a) strings with variable length, (b) search technique using building blocks, where genes in a string are randomly deleted in order to find building blocks containing only good genes, (c) threshold selection in order to ensure that strings only compete when containing some genes from the same gene locus, and (d) cut and splice operators, used for a messy genetic reproduction. In their work, Wu and Simpson (2001) integrated the messy GA with the hydraulic simulator EPANET for solving the systems hydraulic equations in each iteration. Also constraints for pressure, pipe flow, pump capacity, valve settings and tank flow were considered. The model was applied in the two-reservoir network, also tested by Gessler (Gessler (1985) *apud* (Simpson et al., 1994)) and Simpson et al. (Simpson et al., 1994), and in the New York City Tunnels. The performance of the model in a real WSS in Morocco was also investigated. Results showed that in the two benchmark problems, the messy GA found good solutions when compared with the previous studies, always with a significantly reduced computational effort (Wu & Simpson, 2001). In the real case of Morocco, the messy GA demonstrated a faster convergence to better solutions when compared with a simple GA, both starting with similar initial design solutions (Wu & Simpson, 2001).

Later, Wu and Simpson improved their messy GA and introduced the Self-Adaptive Boundary fast messy GA that was tested in the New York City Tunnels problem (Wu & Simpson, 2002). The algorithm was able to find the same solution as the obtained using the messy GA but with a reduced number of the objective function evaluations (Wu & Simpson, 2002).

Abebe and Solomatine (1998) integrated the simulator EPANET with a global optimisation tool (GLOBE) composed by various search algorithms including: controlled random search (two distinct versions, CRS2 and CRS4), genetic algorithm (GA) and adaptive clustering covering with local search (ACCOL). The authors handled the constraints grouping them into hydrodynamic, minimum head and commercial constraints. The hydraulic simulator automatically deals with hydrodynamic constraints, however, for the minimum nodal head violations, penalty functions are applied. The commercial constraints are directly related to the available pipe sizes (discrete space search).

The programme developed by Abebe and Solomatine (1998) was tested with the two-loop and the Hanoi networks. The results demonstrated, as in previous works, the good performance of the GA dealing with this kind of problem. The ACCOL also demonstrated a good performance in both tested networks of the literature.

A comparison between Ant Colony Optimisation (ACO) and Genetic Algorithms applied to the design optimisation of WSS is provided by Maier et al. (Maier et al., 2003). In their model, ACO is linked with the hydraulic solver Wadiso. The ACO has the advantage of consider more available pipe sizes (a larger search space) due to the binary strings dimension. The algorithm has also an improved search into regions where good solutions have been found before.

The ACO demonstrated a similar performance to the GA of Simpson et al. (1994) in the two-reservoir network and a slightly better performance in the New York City Tunnels, obtaining a feasible

global solution (Maier et al., 2003).

Zecchin, Maier, Simpson, Leonard, and Nixon (2007) provide a comparison between five ACO algorithms applied to the two-reservoir problem, to the New York City Tunnels, to the Hanoi problem and to a doubled New York Tunnels problem (2NYTP), consisting of two New York Tunnels networks connected via the single reservoir. The compared algorithms were: (1) Ant System (AS), the original and simplest ACO (Dorigo et al. (1996) *apud* Zecchin et al. (2007)), (2) Ant Colony System (ACS), that adds probabilistic rules to determine whether an ant is to act and also present a ”local” updating of the pheromone, encouraging the exploration of alternative edges (Dorigo & Gambardella (1997) *apud* Zecchin et al. (2007)), (3) Elitist Ant System (ASelite), consisting in the use "elitist ants" in order to maintain the global-best paths after each iteration (Dorigo et al. (1996) *apud* Zecchin et al. (2007)), (4) Elitist-Rank Ant System (ASrank), that includes a rank-based updating at each iteration (Bullnheimer, Hartl, and Strauss (1997) *apud* Zecchin et al. (2007)), and (5) Max-Min Ant System (MMAS), that was developed to solve premature convergence by encouraging local search around the best solution found in each iteration (Stutzle & Hoos (2000) *apud* Zecchin et al. (2007)). The performance of these five ACO algorithms was also compared to other techniques previously applied in the same networks, some of them presented in this paper.

In the case of the New York City Tunnels and the Hanoi network, previous studies have reached lower values for the cost function (Savic & Walters, 1997; Wu & Simpson, 2001), however Zecchin et al. (2007) stated that those solutions have revealed to be infeasible when analysed by EPANET 2.0. ASrank produced the better average performance for the NYCT and MMAS demonstrated to be the best performing algorithm for the Hanoi network (Zecchin et al., 2007).

For the two-reservoir network, although both ACO algorithms reached the global optimum, the ASelite and ASrank demonstrated higher efficiency (Zecchin et al., 2007) even when compared with the best value obtained from other studies (Maier et al., 2003). The MMAS is referred to present the best performance for the 2NYTP (Zecchin et al., 2007).

Globally, the ASrank and the MMAS presented consistently good performances in both case studies standing out from the others algorithms (Zecchin et al., 2007).

Ostfeld and Tubaltzev (2008) applied an ACO algorithm linked with EPANET for the Anytown least-cost design and operation. The design variables considered in this work were the pipe diameters, the pumping maximum power and the tanks storage. Domain pressures at the consumer nodes, maximum amount of water allowed from the source and tanks storage closure were treated as constraints. The Anytown network was slightly modified by an additional source connected to node 9 and a tank to node 4 (see Figure 5.1e). The objective function, in this optimisation problem, includes not only pipe construction cost but also operational and construction cost of pumps and tanks. The proposed algorithm scheme is based on Dorigo et al. (Dorigo et al. (1996) *apud* Ostfeld and Tubaltzev (2008)) and Maier et al. (2003) with some modifications. The only restrictions pointed out by the authors for their methodology are: the fact of the pumps efficiency being considered constant, reliability improvements and fire flow requirements not considered and the use of linear penalty functions instead of more sophisticated constrained handling mechanisms.

Liong and Atiquzzaman (2004) proposed an algorithm called Shuffled Complex Evolution (SCE) for the design optimisation of a WSS. The SCE, developed by Duan et al. (Duan et al. (1992) *apud* Liong and Atiquzzaman (2004)), consists on a synthesis of four successful concepts in global optimisation: (1) combination of probabilistic and deterministic concepts, (2) clustering, (3) systematic

evolution of a complex of points covering the search space in direction to a global improvement and (4) competitive evolution. The global improvements are made through the modification of points from each complex using the Nelder and Mead Simplex method (Nelder & Mead, 1965). Pressure head constraints were treated with penalty cost functions based on the degree of pressure head violation.

The methodology proposed by Liong and Atiquzzaman (2004), coupling the SCE algorithm with EPANET, was tested in the two-loop and Hanoi networks. For the two-loop network, the results were the same obtained by other techniques applied before (GA, GLOBE, Simulated Annealing and Shuffled Frog Leaping Algorithm). However, these results were obtained with a considerable lower number of function evaluations. In the Hanoi case, the optimum obtained by the SCE algorithm was also reached in a considerable reduced computational effort.

M. C. Cunha and Ribeiro (2004) proposed the use of Tabu Search algorithms for the optimal design of a WSS. Two configurations of this kind of algorithm were tested in five examples of network including the two-loop, the New York City Tunnels and the Hanoi. The proposed algorithms were able to obtain identical solutions to the best ones found in previous works for the two-loop and the Hanoi networks. In the case of the New York City Tunnels, the Tabu Search technique was able to find a solution with the same cost obtained by Savic and Walters (1997). For the remaining two cases (one containing a single tank, another with two tanks and both of them with multiple loops), improvements were achieved, demonstrating that the number of benchmark networks being tested in the literature is not enough to conclude about which meta-heuristic algorithm is the most appropriate for this kind of problem.

Geem (2006) developed an algorithm called Harmony Search (HS) which was connected to EPANET for the optimisation of the following networks: (i) two-loop, (ii) Hanoi, (iii) New York City Tunnels and (ii) two distinct networks from South Korea. Cost penalties were applied in the case of constraints violation (Geem, 2006). In the two-loop network, the results were identical to the obtained by the SCE algorithm of Liong and Atiquzzaman (2004). For the New York City Tunnels, the HS was the algorithm that demonstrated the best performance, presenting the lowest cost for the network with a significantly reduced CPU time when compared with the methods mentioned before. The solution obtained for the Hanoi case was the same obtained by Cunha and Sousa using the SA (M. C. Cunha and Sousa (1999) *apud* Geem (2006)) and by Cunha and Ribeiro using the Tabu Search approach (M. C. Cunha & Ribeiro, 2004). In the case of the networks in South Korea, the HS demonstrated a similar performance compared to a GA and a better performance than an algorithm based on NLP (Geem, 2006).

Later, in 2009, Geem tested the HS algorithm in a network containing a pump (Geem, 2009). The objective function included not only pipe capital costs but also capital and energy costs related to the pump. Also a penalty function, to solve pressure head constraints, was added for infeasible solutions proportional to the distance away from the feasible solution area. The same problem was solved by Costa, Medeiros, and Pessoa (2000) using Simulated Annealing. The HS model found the same solution as the SA. However, HS demonstrated to converge faster in finding the optimal solution (Geem, 2009).

Particle Swarm Optimisation (PSO), developed by Kennedy and Eberhart (1995), was tested in 2008 by Montalvo, Izquierdo, Pérez, and Tung (2008) to optimise the design of the New York City and the Hanoi networks. Although the good results obtained, the Harmony Search algorithm of Geem (2006) was still presenting the best performance for both network benchmarks.

A modified harmony search algorithm incorporating particle swarm concept (Particle-swarm harmony search, PSHS) was proposed by Geem (2009). This hybrid concept was tested in three network benchmarks: two-loop, Hanoi and NYCT. The improved HS demonstrated to converge faster than the simple HS, however, it might converge too early. In the case of the two-loop and the Hanoi networks, the PSHS found the same optimum obtained with the simple HS, however faster. In the NYCT, PSHS reached the solution also quickly; however, the cost was slightly higher than the obtained with HS (Geem, 2009).

Differential Evolution was also applied in this kind of problem. Vasan and Simonovic (2010) tested the algorithm in the Hanoi and New York City problems, using EPANET 2.0 for the hydraulic evaluation in each iteration. In the Hanoi case, the same solution obtained previously with HS (Geem, 2006) and with Tabu Search (M. C. Cunha & Ribeiro, 2004) was found in a reduced number of function evaluations. For the NYCT, the DE algorithm took a significantly higher function evaluations number and only reached the same solution obtained with the PSHS algorithm (a near-global solution).

Recently, Bragalli, D'Ambrosio, Lee, Lodi, and Toth (2012) optimised the Hanoi and the New York City water networks using a Mixed Integer Non Linear Programming formulation (MINLP). The authors resorted to Bonmin (*COIN-OR: Bonmin Homepage*, 2011), an open source C++ code for solving general MINLP problems, with a few modifications. The developed formulation allowed a faster achievement of the global optimum in the Hanoi case and the lowest cost function found for the NYCT. Furthermore, the configuration of the solutions obtained by the approach of Bragalli et al. demonstrated to be ready for immediate use in practice, providing a correct hydraulic operation of the networks and a beneficial effect on water quality (Bragalli et al., 2012). This characteristic does not usually occur in designs obtained by some meta-heuristic algorithms based in probabilistic approaches.

Some researchers have verified in practice that the constraints should not be so restricted and neither treated with penalties which can difficult the optimisation process. Instead of that, they claim that some constraints should be treated as optimisation criteria, leading to the concept of multi-objective optimisation.

The Anytown problem was solved through a multi-objective approach, by Walters, Halhal, Savic, and Ouazar (1999), using the Structured Messy Genetic Algorithm, firstly introduced by Halhal, Walters, Ouazar, and Savic (1997). The methodology of Walters et al. (1999) considered two objectives: minimisation of costs and maximisation of benefits resulted from a certain solution (these benefits were evaluated in terms of pressure and storage deficits reduction). In their approach, pumping and storage were included in the optimisation problem. The authors presented two selected feasible solutions (the cheapest and the preferred in terms of operational performance of the network) which demonstrated to be better than any previously published solutions for this specific problem.

Cheung et al. (2003) provide a comparative study between the non-elitist Multi-Objective Genetic Algorithm (MOGA) and the elitist Strength Pareto Evolutionary Algorithm (SPEA). The two-reservoir problem was used to test the performance of distinct algorithms and the hydraulic evaluation was guaranteed by EPANET 2.0. In this work, the considered objectives were costs and pressure deficits minimisation. SPEA demonstrated to be faster than MOGA, requiring smaller processing time (Cheung et al., 2003). The best solution obtained using the multi-objective approach was even lower than the minimum obtained through single-objective approaches.

Formiga, Chaudhry, and Vieria (2006) tested the fast elitist Non-dominated Sorting Genetic Algorithm (NSGA-II), suggested by Deb, Agrawal, Pratap, and Meyarivan (2000), in the two-loop case with some modifications of the original problem including the use of the Darcy-Weisbach formula instead of the Hazen-Williams and the consideration of leakages in pipes. In this case, three objectives were considered for the optimisation of the WSS: minimisation of costs and leakages and maximisation of the reliability (represented by entropy and resilience). The best solution found was 7.4 % superior to the lowest values obtained in previous works, although it might be due to the modifications at the original problem. Nonetheless, the algorithm demonstrated to be capable of find a well-defined Pareto front in a little more than fifty generations.

Farmani et al. (2006) went further and tested the Anytown problem with an Evolutionary multi-objective optimisation method including pump operation schedules in the problem (for a 24-hour period). Their approach included the maximisation of the reliability (resilience only) and the minimisation of costs and residence time (to meet water quality standards). The function for the total cost was defined as the sum of pipe and tank capital costs and pumping operating costs. The cheapest solution found was superior to the obtained previously by Walters et al. (1999) using also a multi-objective approach. However, it has to be noticed that the conditions considered in both cases, such as the number of objective functions, were distinct, which affect the results. Anyway, the fact of Farmani et al. (2006) have considered simultaneously design and operation parameters, as well as included cost, reliability and water quality as objective functions, allowed the achievement of high quality solution networks capable of operate under five loading conditions.

Perelman, Ostfeld, and Salomons (2008) proposed the application of the Cross Entropy (CE) methodology for multi-objective optimisation, firstly introduced by Rubinstein (1999). This methodology incorporates elements from multi-objective evolutionary algorithms and makes use of generated elite solutions for the CE probabilities update instead of use the values of best-fitness functions (Perelman et al., 2008). The approach was tested in the NYCT problem and was capable of reach near-optimal solutions with zero maximum pressure deficits.

Olsson, Kapelan, and Savic (2009) investigated three distinct algorithms based in the Building Blocks strategy for multi-objective design of WSS: (1) Univariate Marginal Distribution Algorithm (UMDA), (2) Hierarchical Bayesian Optimisation Algorithm (hBOA) and (3) Chi-Square Matrix (CSM). The building block identification was one of the strategies tested before by Wu and Simpson (2001) in a single-objective approach (also cited above). The NSGA-II, introduced by Deb et al. (2000), was also used in this work and compared jointly with the other three algorithms. The NYCT and the Anytown were the benchmark problems tested for this comparative study. In the NYCT case, only the NSGA-II and the UMDA were able to find the minimal cost solution (zero cost due to no duplication of any pipe). The hBOA and the CSM presented poor Pareto front coverage. The best zero deficit solution was found by the NSGA-II (Olsson et al., 2009). Respecting to the Anytown case, the UMDA and CSM presented the lowest solutions with zero deficits. Olsson et al. (2009) concluded that the loss of front coverage for small problems make the building blocks identification algorithms unsuitable. However, for large problems, these algorithms outbalance the coverage problem and offer serious advantages over the NSGA-II (Olsson et al., 2009).

To finalise, it should be pointed out that some studies, and especially the ones applying classic algorithms, tend to oversimplify the problems in order to make possible the application of several optimisation techniques in distinct WSS. However, it is very important to focus not only on the per-

formance of the optimisation algorithms but also, primarily, on the details of the design problem without forgetting essential elements in order to always ensure the proper operation of the networks.

D. Kang and Lansey (2011) provide a study that demonstrates the importance of not only consider the transmission mains when optimising the design of a WSS but also include the distribution mains. The authors verified that the simplification of the systems ignoring the local distribution pipes results in oversized optimised systems, followed by excessive pressures. The only difficulty of including local distribution pipes is the increase of the problem complexity. However, to solve this, D. Kang and Lansey (2011) propose a methodology that consists on fixing the local pipes size at their minimum allowable diameters. This strategy allows the inclusion of local distribution pipes in the models without the increase of the number of decision variables.

Respecting to the performance of the optimisation algorithms, D. Kang and Lansey (2011) also proposed an approach to improve the convergence of GAs in order to obtain good solutions with much less computational effort. Their approach is based in the generation of logical initial populations using engineering judgement instead of the typical random generation. Results showed a consistent optimised pipe layout using this heuristic approach and an inconsistent optimised layout using the random generation.

5.2 Operational optimisation

The WSS control operations can be included in design optimisation problems. However, due to the burden of operational costs in the total cost of a WSS, the control optimisation can emerge as a particular optimisation problem.

The optimisation of the WSS operation consists in find the best strategies for the control elements minimising the total costs while satisfying the consumers demand in terms of flow and pressure conditions (Cembrano, Brdyś, Quevedo, Coulbeck, & Orr, 1988). In several scientific works, the control optimisation problem is treated as a single-objective problem consisting in the minimisation of the operational costs and the use of constraints to satisfy the WSS requirements. However, other works look into this kind of problem as a multi-objective optimisation problem: minimisation of costs and maximisation of hydraulic benefits (Carrijo, Reis, Walters, & Savic, 2004; Savic & Walters, 1997), in resemblance to what was shown previously for design optimisation.

A control optimisation strategy can also be static or dynamic when real-time systems are used simultaneously (A. A. R. Cunha, 2009). Real-time control approaches are discussed in section 5.2.2. Control optimisation models have been proposed since the seventies, exploring several optimisation techniques such as the most traditional (i) Linear Programming (Firmino, Albuquerque, Curi, & Silva, 2006; Vieira & Ramos, 2008, 2009) and (ii) Non-linear Programming (Brion & Mays, 1991; Cembrano et al., 1988; Cembrano, Wells, Quevedo, Pérez, & Argelaguet, 2000; El Mouatasim, Ellaia, & Al-Hossain, 2012; Vieira & Ramos, 2008), but also the meta-heuristics derived from nature such as (i) Genetic Algorithms (Carrijo et al., 2004; Mackle, Savic, & Walters, 1995; Rao & Salomons, 2007; Savic & Walters, 1997; Shihu et al., 2010), (ii) Simulated Annealing (Goldman & Mays, 2005; Shihu et al., 2010), (iii) Ant Colony Optimisation (López-Ibáñez, 2009), etc. Genetic Algorithms and mainly Hybrid Genetic Algorithms (Shihu et al., 2010; Van Zyl, Savic, & Walters, 2004) have standing out for their strong ability to solve optimisation problems with high level of non-linearity and also for their performance dealing with the multi-objective optimisation perspective.

In the literature, the most used water networks benchmarks for control optimisation are mainly: (i) the case-study of Van Zyl, (ii) the Richmond network in UK, (iii) the network of the city of Austin in Texas and (iv) the North Marin Water District in California.

5.2.1 Pumping systems

In WSS, pumping energy costs usually represent the main costs of the water companies (Van Zyl et al., 2004; Vieira & Ramos, 2008). Pumping systems represent nearly 20 % of the world's energy used by electric motors and 25 to 50 % of the total electrical energy required in some industries (Europump and Hydraulic Institute, 2004). All these facts imply an increasing demand to control pumps efficiently by the water industry.

Inefficient pumps, inefficient pump combinations and inefficient pump scheduling are the three main problems that are commonly found in pump stations. Thus, the use of optimisation techniques for the improvement of the pumping systems is crucial, either for the optimal schedules computation or for the optimal combinations.

Pumps can be controlled according to variations in suction pressure or even by time controls (Feldman, 2009). However, in most of cases, pumps are controlled by the reservoirs water level variations. In these cases, pumps are only switched on when the reservoirs responsible for supply certain populations are empty (or in the minimum level) and switched off when the same reservoirs reach the maximum level allowable. If the pumps operated according to the variation of the energy tariff during a day and according to the water consumption patterns, then the associated costs would been significantly reduced (Coelho & Andrade-Campos, 2012).

When optimising the pumps operation it is possible to obtain not only energy savings but also better performances, improved reliability and even reduction in life cycle costs (Europump and Hydraulic Institute, 2004).

Nowadays, there are a large number of scientific works dealing with operational pump optimisation, usually referred as pump scheduling optimisation (López-Ibáñez, Prasad, & Paechter, 2008; Savic, Walters, & Schwab, 1997; Van Zyl et al., 2004).

In the literature, there are essentially two kinds of explicit pump schedule optimisation problems: (i) the most common deals with constant speed pumps, where only two solution variables are considered for the pump operation (with the values 1 or 0, usually representing the pump status switched on or switched off), and (ii) the other deals with variable speed pumps, where the values of the optimisation variables are defined by the set of speeds of the pump. Some works also investigate the operational optimisation of the pumping systems through the perspective of other elements of the network (implicit formulation), where the decision variables can be represented by the reservoir levels variation, the pump station discharge, the supply pressure or the time of pumps operation (Ormsbee, Lingireddy, & Chase, 2009).

Typically, to solve this kind of problem, simulation periods of 24 hours with 1-hour time-steps are used. However, Bene and Hős (2011) had demonstrated with their study of least-cost filling reservoir that the choice of time-steps smaller than 1 hour can provide better results. The main conclusion of their work was that by fixing a temporal time-step and sequentially setting the pump operating point (minimal energy consumption), a globally reservoir filling policy can be realised. However, the technique loses its optimality if the energy tariff or the consumption changes during the optimisation

process (Bene & Hős, 2011).

Ormsbee et al. (2009) provided three mathematical formulations for pump scheduling to minimise the associated energy costs, described by: (1) an implicit formulation, where the decision variables can be the pump stations discharge, the tanks water levels or the pressure and then, the pump schedule associated to the obtained solution need to be found; (2) a discrete explicit formulation, where the number of decision variables is given by the product between the number of pumps and the time intervals in which each pump operates (these time intervals can be restricted or unrestricted, *i.e.* can be restricted to some periods of the simulation time horizon or can include all the time horizon); and (3) a composite explicit formulation, in which is attributed a single decision variable for each pump station. These formulations can be solved through unconstrained methods applying penalties or through constrained methods where the constraints can be directly incorporated in the algorithm (Ormsbee et al., 2009).

The gradient-based optimisation methods were the first being tested for pump scheduling optimisation (Brion & Mays, 1991; Cembrano et al., 1988) and then, the nature-based algorithms became to emerge (Goldman & Mays, 1999; Mackle et al., 1995; Savic & Walters, 1997), demonstrating to be more adequate since there is no need of oversimplification of the problems using. In the last ten years, a large variety of studies applying a number of distinct meta-heuristic algorithms combining global and local search techniques, which tend to improve the convergence of the methods, have emerged.

The works of Cembrano et al. (1988) and Brion and Mays (1991) are examples of well-succeed applications of classic algorithms for the pump schedules optimisation.

Cembrano et al. (1988) tested the Conjugate Gradient method in the Barcelona network (Spain) using a single-objective approach to minimise operational costs considering as decision variables the flow combination provided by the pump stations, valves or turbines (continuous variables). Although the Barcelona network is composed of 4 sources, 5 valves, 7 pump stations (of fixed-speed), 2 turbines, 11 demand areas with distinct pressure zones and 11 storage reservoirs, a linear model for the dynamic behaviour of the system was considered. State and boundary constraints were considered by the authors and penalty functions were used in the case of constraints violation. Their model was able to find optimal schedules for the pumps and optimal valves control capable of reducing the operational costs associated. The optimised operational results obtained were similar to the obtained in a previous work using Dynamic Programming. However, the computational effort required by their latter methodology was lower (Cembrano et al., 1988).

Brion and Mays (1991) resorted to KYPIPE computer programme to optimise the operations of the Austin network (Texas) which comprises 1 pump station with 3 parallel fixed-speed pumps, 8 pressure zones and 2 storage reservoirs. The model applies a Generalised Reduced Gradient algorithm and an Augmented Lagrangian method for the application of penalties. Considering a time horizon of 24 hours divided into 12 2-hour time-steps for the pump schedules, Brion and Mays obtained a reduction in the Austin operational costs of 17.3 % (Brion & Mays, 1991). The authors verified that the algorithm used were very sensitive to the Lagrangian coefficients.

The network of the city of Austin was later optimised by Goldman and Mays (1999) using Simulated Annealing (SA). Constraints to the tanks levels, to the nodal pressures, continuity and water quality were taken into account in their optimisation process. The pump operational costs were reduced 4.1 % taking twice the number of iterations required by a Non-linear Programming (NLP) method. The costs reduction was quite small, however, it should be noticed that additional constraints

were included. At the same time, SA demonstrated to be more flexible and adaptable than NLP, providing a number of optimal pump schedules instead of only one (Goldman & Mays, 1999). The same model was also applied to the North Marin Water District network (California), comprised of 2 sources, 2 fixed-speed pumps and 3 tanks, and demonstrated once again a good performance in finding optimal pump schedules (Goldman & Mays, 1999).

The good performance of the Genetic Algorithms (GA) applied to the operational optimisation of a WSS was soon demonstrated by Mackle et al. (1995). The authors tested this algorithm in a simple network containing one source, four parallel fixed-speed pumps and a single storage reservoir. Considering continuity and reservoir water levels constraints, an optimised pump schedule adapted to the energy tariff was obtained in just 20 minutes (10000 generations).

The network of Richmond (UK) has been subject of many control optimisation studies (Atkinson, Van Zyl, Walters, & Savic, 2000; López-Ibáñez et al., 2008; López-Ibáñez, Prasad, & Paechter, 2011; Van Zyl et al., 2004). This benchmark network, presented in Figure 5.2a, is composed of a unique water source, a pump station containing two parallel fixed-speed pumps, five booster pump stations (with a single pump each), six tanks and distinct pressure zones.

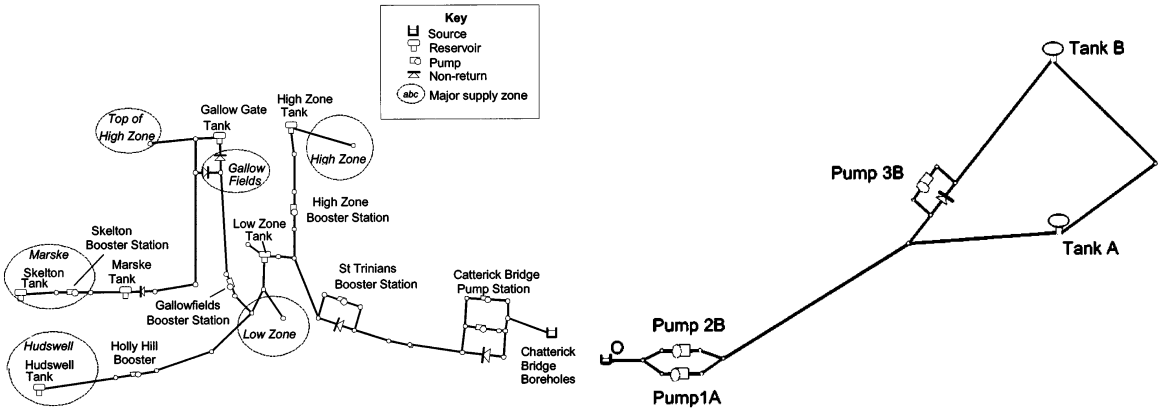


Figure 5.2: Representation of the most tested benchmark water supply networks for control optimisation: (a) the Richmond water network and (b) the Van Zyl network case- study (both adapted from Van Zyl et al. (2004)).

In 2000, Atkinson et al. (Atkinson et al. (2000) *apud* Van Zyl et al. (2004)) reduced in 19 % the operational costs of the Richmond network using a commercial hydraulic simulator and GAs. However, due to the burden of the simulator, the computational time was extremely high (69 hours).

Later, Van Zyl et al. (2004) developed a hybrid optimisation strategy, combining a Genetic Algorithm with two types of Hill Climber methods: (i) Fibonacci coordinate search and (ii) Hooke and Jeeves pattern search. As the GA is efficient in identifying the region of the optimal solution but less efficient in finding the optimal point in that region, the introduction of a technique like the Hill Climber can improve the local search. In this study, the optimisation variables were defined in terms of tank level controls. A pump penalty cost and a tank penalty cost were used to impose the system constraints (tank water levels and number of pump switches). These penalties were determined by a trial and error approach. Applied to the Richmond network, the hybrid method proved to be superior to the pure GA, reducing more than 25 % of the operational costs. The developed methodology was

also applied to a particular case study (see Figure 5.2b), introduced by the authors, comprising 1 water source, 1 pump station (2 parallel fixed-speed pumps), 1 booster pump station (1 pump) and 2 tanks. Also in this case, the hybrid GA performed better than the pure GA both in convergence speed and quality of solutions (Van Zyl et al., 2004). The best results obtained for the hybrid algorithm were with the Hooke and Jeeves technique. The greatest difficult in this work, as referred by the authors, was to decide when to change from the GA to the Hill Climber method. This is a typical difficult task to solve for those who work with more than one optimisation algorithm.

Comparing their work with the previous published by Atkinson et al. (Atkinson et al. (2000) *apud* Van Zyl et al. (2004)), Van Zyl et al. stated that the same reduction of 25 % could be obtained using the methodology of Atkinson et al. if the constraint for the 95 % full requirement of the tanks have been applied at 7:00 instead of 5:00.

Trying to reduce the computational effort of the optimisation problem in the Richmond network, López-Ibáñez et al. (2008) proposed the use of a parallel Ant Colony Optimisation (ACO) technique instead of the common sequential technique, *i.e.* instead of a sequential iteration between ants during the optimisation, the authors propose the iteration through some threads in parallel. EPANET library was then combined with a parallel ACO algorithm. Results of the application of this methodology in the Richmond network showed that the optimal solution was found in 8000 iterations (the same required by the hybrid GA of Van Zyl et al. (2004)) consuming less than half an hour, while using a sequential technique would take around two hours. The authors also verified that, in the parallel ACO, a higher number of ants reduces computation time, which is the opposite in the sequential ACO. Another advantage of the high number of ants is the possible improvement of the final solution.

A study comparing three distinct representations of the decision variables for fixed-speed pumps is provided by López-Ibáñez et al. (2011), in which both methodologies were tested in the case-study of Van Zyl et al. (2004) and also in the Richmond network. The authors compared (i) the common binary representation, (ii) the level-controlled triggers and (iii) the time-controlled triggers representations using the Simple Evolutionary Algorithm (SEA) linked to EPANET. The violation of the constraints related to tanks water levels, continuity, nodal pressure head, number of pump switches and to the occurrence of warnings on EPANET were handled by a method based on ranking solutions with respect to the constraint violation (López-Ibáñez et al., 2011). Globally, the SEA with time-controlled triggers demonstrated better performance when compared to the SEA with both binary and level-controlled triggers or even compared to the Hybrid GA of Van Zyl et al. (2004), also using level-controlled triggers. It should be worthy noticed that López-Ibáñez et al. (2011) believed that adapting the hybrid GA of to the time-controlled triggers may further improve the results.

The case study of Van Zyl et al. (2004) had also been analysed by López-Ibáñez, Prasad, and Paechter (2005) using a multi-objective approach with the SPEA2, a second version of the Strength Pareto Evolutionary Algorithm coupled with EPANET 2.0. Their model conjugates a number of improvements that had been implemented in other works using multi-objective approaches. The objectives considered on this model were the minimisation of both costs and number of pump switches. Constraints for tanks water levels and nodal pressures were handled by a method based on ranking solutions with respect to their constraint violations. Results demonstrated that, considering a 24-hour horizon (1-hour time-steps) and using a binary encoding of the decision variables, better solutions are obtained when compared to the same approach for a single-objective optimisation (López-Ibáñez et al., 2005).

Other works have also been presenting the performance of other optimisation algorithms and innovative techniques. However, it is notorious the lack and, consequently, the need for comparison of some of these methodologies in benchmark networks in order to obtain more valuable results.

For a case study in Brazil (Goiania network), Carrijo et al. (2004) used the SPEA connected to EPANET 2.0 for the minimisation of operational costs (pumps and valves) and the maximisation of hydraulic benefits (evaluated through indexes for pressure requirements, demands and reservoir levels). In this work, it is also introduced the use of data mining to extract operational rules for the system in order to reduce the dependence on experts for the choice of the most adequate solution in the obtained Pareto front.

The work of von Lücken, Barán, and Sotelo (2004) offers a comparative study using sequential and parallel implementations of six distinct Multi-Objective Evolutionary Algorithms (MOEAs): (1) Multiple Objective Genetic Algorithm (MOGA); (2) Niched Pareto Genetic Algorithm (NPGA); (3) Non-Dominated Sorting Genetic Algorithm (NSGA); (4) Strength Pareto Evolutionary Algorithm (SPEA); (5) NSGA-II; and (6) Controlled Elitist NSGA-II (CNSGA-II). The objective of the work consisted on the minimisation of both (i) energy cost, (ii) number of pump switches, (iii) maximum power peak and (iv) reservoirs levels variation (von Lücken et al., 2004). Parallel implementation of MOGA, NPGA and NSGA do not find any non-dominated Pareto solution. On the other hand, the best position was obtained by the parallel implementation of CNSGA-II using 16 processors (von Lücken et al., 2004). The authors identified several improvements of their parallel strategy over sequential MOEAs (von Lücken et al., 2004): (i) exploration in larger areas due to a larger number of populations; (ii) introduction of cooperation between populations which helps the search for good solutions; and (iii) inclusion of a process for elitism reinforcement, preserving solutions that could be lost.

J.-Y. Wang, Chang, and Chen (2009) were the first researchers considering the land subsidence due to groundwater pumping all day long, which can be solved using intermittent pumping. In their model, a Genetic Algorithm is used for the minimisation of both pump operational cost, number of switches and total work time for each pump. Constraints to control the reservoir levels and the flow in the system, to avoid underflow or overflow, were handled with penalty functions. The model also includes local search for the improvement of the solution quality and the pump control was performed by time interval representation using a real-number array instead of a binary bit string. A number of possible solutions with lower electricity cost and, at the same time, with eco-aware schedules were achieved. However, the authors considered that the convergence speed could be improved (J.-Y. Wang et al., 2009).

Firmino et al. (2006) used a two-stage optimisation method based on Linear Programming and Integer Linear Programming by an optimisation toolbox of MATLAB 7. This method was applied to Campina Grande WSS in Brazil, containing three pumping stations, and has saved around 15 % of the costs and energy consumption (single-objective approach). Constraints to the reservoirs levels, to the maximum allowable pump flow and for the guarantee of periodicity of the schedules (continuity) were considered.

A case study in China (Shihu et al., 2010) of a large-scale WSS, containing fixed- and variable-speed pumps, demonstrated a reduction on energy costs of 6.04 % using a Hybrid Genetic Algorithm called Genetic Simulated Annealing (GSA) for the optimisation of the pump schedules. The developed methodology considers a single objective function which includes not only electricity cost of

pumping but also the water production cost. Constraints to the tanks water levels, continuity, velocity limits of the variable-speed pumps and also to the number of pump switches were considered (Shihu et al., 2010). It has to be noticed that this problem deals both with fixed- and variable-speed pumps, which implies distinct decision variables and increases the complexity of the optimisation problem. At the same time, a good algorithm for the optimisation of fixed-speed pump schedules could not be adequate in the case of variable-speed pumps. This topic was not yet analysed in detail in previous works and it is of large importance since most actual WSS can be constituted by distinct kinds of pumps.

Recently, El Mouatasim et al. (2012) proposed the use of a reduced gradient algorithm for the problems of Kanitra and Agadir cities in Morocco. Their methodology based on the algorithm called Stochastic Perturbation of Reduced Gradient (SPRG) was compared with the optimisation solver LINGO. The objective function considered is the sum of the cost function of every wells and treatment plants existent in the networks. The pumps were only allowed to operate in three time periods of the 24-hour horizon. Results for both networks indicated a significant better performance of the SPRG compared to the LINGO solver (El Mouatasim et al., 2012).

Also Coelho, Tavares, and Andrade-Campos (2012) provided a work comparing distinct optimisation algorithms applied to water networks evaluated using EPANET 2.0. The authors tested three distinct algorithms: (i) an Evolutionary Algorithm (EA), (ii) a gradient-based algorithm and also (iii) an hybrid algorithm, called HDEPSO (Caseiro, Valente, Andrade-Campos, & Yoon, 2011), which combines Differential Evolution (DE) with Particle Swarm Optimisation (PSO). The operation of variable-speed pumps during a time horizon of 24 hours was optimised taking into account the water consumption of the population and the variation of the energy tariff.

5.2.2 Real-time operations

A Supervisory Control and Data Acquisition (SCADA) is a system also used in Water Supply Systems for the real-time control and monitor of several elements such as pumps, valves, reservoirs, etc. (Walski et al., 2003). Real-time strategies allow optimal operational adjustments to possible variations in the networks such as sudden fluctuations in demand, contributing for the efficiency improvement of the WSS.

According to Gellings (2009), the potential savings of the use of SCADA systems are from 10 to 20 % of total WSS energy consumption. Becoming the system automatic, efficiency is increased and a reduction on costs occurs.

A SCADA is basically a system composed by one or more field data interface devices such as reservoir level meters, water flow meters, valve position transmitters, power consumption meters and pressure meters (Walski et al., 2003). The all system must incorporate (Walski et al., 2003): (i) a central host computer server (or servers), (ii) a type of communication to transfer data between field data interface devices and the computers of the central host (radio, cable, satellite, telephone, combinations of these or others), and (iii) a software to allow central host and terminal operator applications and to support the communications and the devices. Thereby, the main functions of a SCADA system are (1) data acquisition, (2) data communication, (3) data presentation and (4) control.

Figure 5.3 shows an example of a scheme for the optimal operation of a WSS using a SCADA system. The scheme incorporates modules for water demand prediction, optimisation and hydraulic

simulation. The automated real-time control system allows the data management and the transmission of information between both modules and the WSS that is intended to optimise. A data base is also necessary to record the historic of water consumption and all the characteristics of the real network.

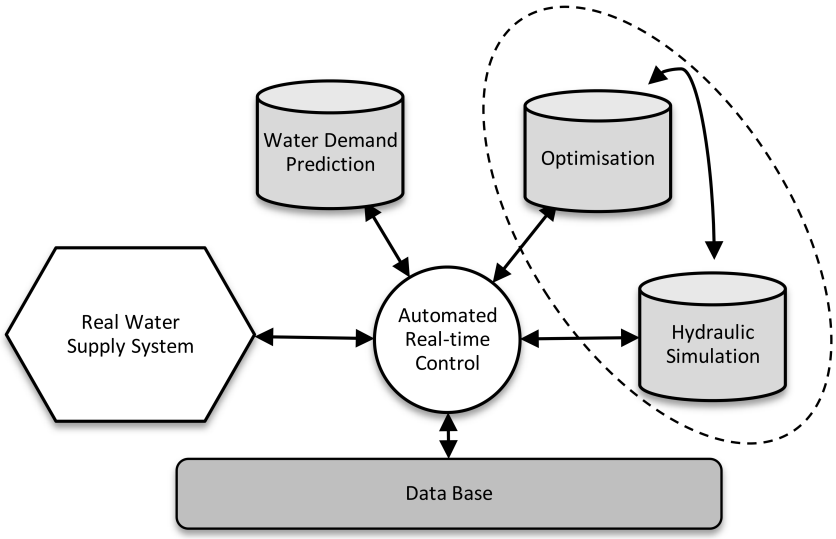


Figure 5.3: Scheme describing the optimal operation of a Water Supply System using a SCADA system.

A large number of works dealing with real-time operational optimisation of WSS were already published (*e.g.* Bunn (2007); Cembrano et al. (2000); Coulbeck, Orr, and Brdyś (1988); Martinez, Hernandez, Alonso, Rao, and Alvisi (2007); Rao and Salomons (2007); Salomons, Goryashko, Shamir, Rao, and Alvisi (2007)). Fallside and Perry (1975) and Coulbeck et al. (1988) were some of the first researchers publishing in this area.

A similar scheme to the represented in Figure 5.3 is presented by Coulbeck et al. (1988), describing with more detail the basic modules (prediction, optimisation and simulation) and procedures for the successful implementation of a real-time completely automated optimal control strategy. The prediction module (called GIDAP, in this case), integrating demand analysis and prediction, involves distinct processes (Coulbeck et al., 1988): (i) screening of the telemetry data (error values are replaced by interpolation or previously predicted values); (ii) data smoothing, to remove possible disturbances; and (iii) trend estimation, for the demand estimation through the use of screened and smoothed data. More details about demand prediction can be consulted in Section 5.3. In the optimisation module (called GIPOS for a single water source and GIMPOS for multiple sources), pumping and storage are controlled by the computation of optimised pump schedules taking into account constraints of the reservoirs water levels, the consumers demand (provided by the prediction module) and also the general constraints related to the system operation (Coulbeck et al., 1988).

The success of the optimisation procedure has a strong dependence on the hydraulic model of the system. In some cases, the system model can be incorporated in the optimisation procedure although it tends to become oversimplified. For this reason, the recourse to a hydraulic simulator is usually desirable for providing a better approximation of the system performance. At the same time, the hydraulic module can be used to compare the actual performance of the real network with the

performance after optimisation and also used for non-routine or emergency situations (Coulbeck et al., 1988).

The typical data transferred by the SCADA systems comprise (Coulbeck et al., 1988): pump and pipe flows, pressures, control states and reservoir levels. System monitoring, usually carried out by means of telemetry, is required to guarantee the correct operation of the network since regulatory actions are dependent on the predicted behaviour of the system under a set of conditions (Coulbeck et al., 1988).

Coulbeck et al. (1988) also refers the importance of a unique data base, which allows the combination between the recent telemetry information and all the other data required for the individual modules.

The real-time control scheme of Coulbeck et al. (1988) was implemented in a network in the United Kingdom providing not only automatic least-cost pump schedules but also instantaneous system assessment and automated system operations. Fallside and Perry (1975) introduced a hierarchical approach for the online optimisation of a water supply network. This hierarchical approach consisted firstly on the use of linked computers, one for optimisation calculations and the others for data measurement and system control (decentralised optimum computer control). Secondly, the optimisation process was based on a hierarchical decomposition technique employing Lagrange duality theory. Basically, the dual of the original problem is formed and then, a decomposition technique allows obtaining a set of smaller and independent problems easier to solve through a standard method.

The methodology of Fallside and Perry (1975) was applied to the East Worcestershire system that operates 7 source stations, 5 spring sources, 15 pressure boosters, 21 service reservoirs and 6 water towers. Pumps were operating in fixed- or variable-speed to maintain the best efficiency in the operating range. The system was also provided of an automatic control consisting on a central computer connected to all major elements of the system by radio or land-line. The control system was able to provide information about 200 elements every 65 seconds (Fallside & Perry, 1975). The developed hierarchical techniques allowed essentially the reduction of the computational burden (Fallside & Perry, 1975).

Cembrano et al. (2000) developed an online optimal control tool using an optimisation solver called WATERNET and other management tools for simulation and quality control, both linked to a SCADA system. The developed user interface allows the easy control optimisation of distinct water networks. The methodology of Cembrano et al. (2000) consisted on the use of a Generalised Reduced Gradient algorithm for the real-time optimal valve and fixed-speed pump controls. The entire control system was then composed by an optimiser, a demand forecast and a SCADA system. This type of control was tested in a prototype of the network of Sintra (in Portugal), resulting on a total cost reduction of around 18 % (Cembrano et al., 2000).

Pegg (2001) showed the implementation of the Derceto's computer programme for the operational optimisation of the Wainuiomata-Waterloo network (Wellington), which comprises 3 water treatment plants, 12 reservoirs, 5 fixed-speed pumps (4 standby), 2 variable-speed pumps (1 standby), 1 dual-speed pump (standby) and 1 control valve. The programme was set to run every half-hour and previous solutions were maintained until a new one was provided.

For the real-time control, Pegg (2001) implemented Derceto's programme in a computer, running WindowsNT and linked to the control system responsible for provide telemetry information. The selected operator interface was a SCADA system called Citect (for more information about Citect see

Schneider Electric (2012)).

The main steps in Derceto's programme operation includes (Pegg, 2001): (1) initialisation of system data, (2) determination of the mass-balance required to get all the reservoirs full in the end of the day, (3) computation of the lowest cost schedules and (4) checking of the results using the simulator EPANET. These stages typically run several times to improve the accuracy and then, the information is passed to the telemetry or central control system.

Pegg's work results demonstrated a reduction on energy cost of the Waterloo network of approximately 10%. During the real-time implementation, some events such as (i) pump failure, (ii) pipe maintenance, (iii) telemetry failure or (iv) systematic errors on metered data have occurred. However, the used programme was able to deal with the unusual situations by a quickly adaptation to the unexpected changes (Pegg, 2001).

Later, a work containing results of the real-time optimisation using Derceto's programme (Aquadapt) in two distinct cases was provided by Bunn (2007). The tested systems were the East Bay Municipal Utility District (EBMUD) and the Washington suburban system. EBMUD already had a centralised pump scheduling package which facilitated the application of Aquadapt, allowing energy cost savings of 13.1 %. The case of the Washington suburban system presented more difficulties in the interface because there was no centralised Programmable Logic Controller (PLC). Thus, the existent Remote Terminal Units (RTUs, also referred as Remote Telemetry Units) were replaced by smart PLC. Results showed energy cost reductions up to $1000/day in the third week of implementation (Bunn, 2007). All these savings were obtained by (i) moving energy use to cheaper periods, (ii) reducing peak demand charges and (iii) reducing the energy required for pumping (Bunn, 2007).

Although PLCs have similar functionality as RTUs, they present the advantage of combine large quantities of digital and analogue data and produce algorithms of high complexity (Bunn, 2007). On the other hand, an RTU usually does not support control algorithms or control loops.

The JEA's Operation Optimisation System (OOS) project presented by Barnett et al. (2004) is another case incorporating demand forecasting, modelling, simulation and optimisation coordinated by a SCADA system. The hydraulic model used in this project was the WaterGEMS and its calibration was through historical samples and real-time data from the Jacksonville water network (Florida, USA). This network, composed of 32 wells, was also used for testing the real-time OOS. The automatic model developed in this project included techniques such as (Barnett et al., 2004): (i) neural networks for consumption prediction, (ii) non-linear constrained optimisation for pump and valve schedules and (iii) mechanistic hydraulic and mass-balance for water supply modelling.

The most significant improvement in the Jacksonville network provided by the real-time optimisation was the capital costs reduction. Energy savings and water quality improvement were also identified and, moreover, the return on investment for this project was less than one year (Barnett et al., 2004).

The POWADIMA research project (Rao & Salomons, 2007) has developed a real-time methodology which combines the use of an Artificial Neural Network (ANN) for predicting the consequences of different pump and valve control settings (hydraulic behaviour) and a GA optimiser for selecting the best controls combination. Constraints to pressures, flow velocities and reservoirs levels were also included. A SCADA system was responsible for providing data updates for each 24 hours. This methodology was applied in the Haifa-A WSS of Mount Carmel, comprising 9 storage reservoirs and 17 fixed-speed pumps (5 pump stations), reducing in 25 % the energy costs (Salomons et al., 2007).

The Valencia WSS (Spain), containing 10 fixed-speed pumps (3 standby), was also tested, indicating an operational cost savings of 17.6 % (Martinez et al., 2007). Comparing the same methodology combined with EPANET for the hydraulic behaviour verification instead of the recourse to the ANN, it was verified, for the Haifa-A case, that the GA-ANN model was approximately 25 times faster than a 112-node GA-EPANET model (Salomons et al., 2007).

Ingeduld (2007) investigated the use of EPANET on real-time operations. According to the author, linking a SCADA system to the hydraulic simulator EPANET allows the water network to operate into the following modes (Ingeduld, 2007): (a) Virtual-Sensor Mode, based on the WSS conditions which provides information about locations without measurements; (b) Hindcasting Mode that allows to obtain simulations of past events in the network; (c) Event-Simulation Mode that provides the response of a system when a specific modification is applied; and (d) Predictive Mode that provides a prediction of the system behaviour for a certain time horizon.

To provide data transfer between the SCADA system and the hydraulic simulator, Ingeduld (2007) pointed out the use of a Data Integration and Management System (DIMS) based on a SQL (Structured Query Language) client-server (see Figure 5.4). The data communication is provided by online hosts and data drivers.

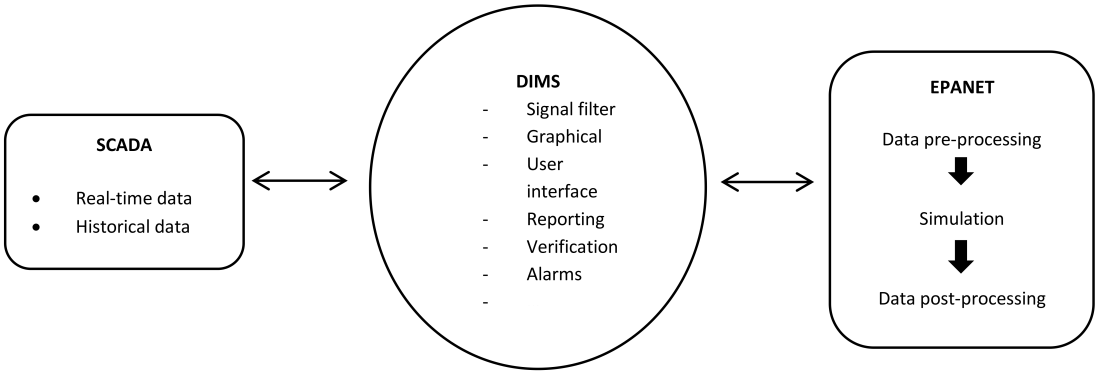


Figure 5.4: General scheme of the online EPANET solution presented by Ingeduld (2007).

The online solution presented by Ingeduld (2007) was applied in two case studies: (i) one at Czech Republic and (ii) another at Libya (specifically in the well-known Great Man-Made River project). In the first case, a SCADA SCX system and the MikeNet (see section 4.2) were used. For the Great Man-River project, a Siemens SCADA system was selected for the data collection. In this second case, updates every 5 minutes and consequent analysis of the system were executed. Furthermore, historical data analysis and prediction of the system behaviour for the following 72 hours were also performed.

The work of Machell et al. (2010) also provides an insight about online modelling of WSS and demonstrates its potential to detect events in the networks like ruptures. The authors used the AQUIS software, an industry standard modelling package in the UK, that receives, every 30 minutes, flow and pressure data in real-time obtained by a GPRS communication. The online model was linked to DataManager, a database for configuring, pre-processing and administering that allowed to check out for missing and corrupt data (Machell et al., 2010). The online model based on a SCADA system, applied to a UK case study, demonstrated to be able to provide early warnings of effects in each pipe

of the networks and information about which customers were affected (Machell et al., 2010).

Dötsch, Denzinger, Kasinger, and Bauer (2010) provided a decentralised approach for the real-time control of a WSS. This approach consists on the communication between equipment (called agents) installed at each pump and tank of the WSS for the minimisation of energy costs associated to their operation. For the self-organised and decentralised coordination of the agents communication, the authors have resorted to a chemical stimuli strategy based on a biological phenomenon applied by plants to keep the herbivores away (Dötsch et al., 2010). Two variants of their approach were compared (Dötsch et al., 2010): (i) Greedy Mode (GM), the simple variant where the solution is based on the switch on and off of the pump nearest to a tank; and (ii) Coordinated Greedy Mode (CGM), a variant where the pump agents compare all possible solutions according to additional expected costs information and choose the most efficient. Experimental results, using EPANET for the hydraulic evaluation of the networks, demonstrated that both variants provide optimal solutions. Moreover, it was also showed that in the CGM, more cost efficient controls can be obtained when additional information is provided to the agents (Dötsch et al., 2010).

In these sections, relating to the influence of the connection between SCADA and optimisation modules, it is worth notice that no investigation was made in the ability of the algorithms to adapt to small changes in constraints conditions such as online changes in predicted demand and current reservoir levels.

5.2.3 Systems with energy production

With the increasing introduction of renewable energy production in WSS, the study of control optimisation techniques including possible new elements such as hydroturbines or wind-turbines is of high importance in order to keep the maximum energetic and economic efficiency of the systems.

The control problems of installations containing renewable energy sources are similar to the problems mentioned before. For pumps operating as turbines, for example, during the normal pump operation, both values of the characteristic curves (flow, head and speed) are positives; however, in the turbine mode, the discharge and speed present negative values (Ramos et al., 2010). Usually, pump manufacturers do not provide the characteristic pump curves when it operates as turbines (Ramos et al., 2010).

Ouarda and Labadie (2001) presented in their work the operational optimisation of a four-reservoir case study of Argentina (Hidronor river basin) containing hydropower plants. The authors used a chance-constrained optimal control resorting to optimal control theory (OCT) that involves the use of ordinary and partial differential equations in continuous formulations (Ouarda & Labadie, 2001). This kind of formulation usually requires less computational effort than the typical mathematical programming (MP) techniques. At the same time, the presented technique is normally used for several continuous-time optimal control problems under constraints (Ouarda & Labadie, 2001).

The main objective of the Ouarda and Labadie work (Ouarda & Labadie, 2001) was the maximisation of the total energy production at hydroelectric plants. For that, two representations of the objective function were considered with the aim of evaluating the performance of the optimisation algorithm with distinct representations. The difference between both representations is that in the first, only the total energy production is considered and, in the second, the energy rate per turbine discharge is also considered (Ouarda & Labadie, 2001). Using the first representation of the objective function,

a local optimum was obtained. The better values were achieved using increasing penalty weights (for the constraints violation) by multiples of five to fifty, leading to final objective values up to 96.4 % of the true global optimum. Using the second representation of the objective function, the algorithm converges for the optimal optimum (Ouarda & Labadie, 2001). The authors also have considered measures for the system reliability, considering a range from 5 to 95 % (high to low theoretical risk) and the results provided optimal values reflecting several levels of reliability.

Teegavarapu and Simonovic (2002) compared the performance of a mixed integer non-linear programming with an improved simulated annealing technique applied to a real-time four-reservoir hydropower system at Canada. The considered improvements of the SA passed through repair strategies to generate feasible solutions for any configuration of the reservoir systems and heuristic rules to define the range of discharge variables (Teegavarapu & Simonovic, 2002). The objective of the problem was to minimise the cost of power generation, considering half-day scheduling and 14 time intervals (1 week). A constraint is applied in order to guarantee that the total power demand by the reservoirs for both SA and MINLP models is the same (43 GWh). At the same time, discharge ranges are defined according to the variation of the generation cost. The SA model provided the best value for the objective function with less computational effort when compared with the MINLP formulation. The higher computational effort required by the MINLP is related to the high number of binary variables proportional to the number of time intervals. For both used techniques, it was also observed that the generation values were significantly higher for periods of lower generation cost (Teegavarapu & Simonovic, 2002).

Gonçalves and Ramos (2008) proposed an optimisation methodology based on an economic analysis for the use of pumps-as-turbines (PATs). The main objective of this work was the identification of regions constituting potential for energy production in the WSS of the district of Aveiro (Portugal). Then, it was intended (i) the evaluation of the available energetic potential, (ii) the calculation of energetic production, (iii) the study of viability and economic analysis and, finally, (iv) the development of the optimisation proposal (Gonçalves & Ramos, 2008).

For the energetic potential evaluation, pressures and flows were analysed in five areas containing pressure reduction valves (PRV). All the hydraulic analyses were performed with the hydraulic simulator EPANET. The estimation of the energy production was based (i) on the headloss measured at each PRV using EPANET and (ii) on the demand prediction considering a population growth during 10 years (Gonçalves & Ramos, 2008). Concerning to the economic analysis, for the possible flows and powers to install, the authors considered a criteria based on the Net Present Value (NPV) determined by the difference between the annual profits and the annual costs related to installation and maintenance, which allow increasing the profitability of the systems. For the optimisation of the system operation, considering the real-time consumption patterns and reservoirs levels variations, the authors proposed the use of a multi-objective genetic algorithm (MOGA) connected to EPANET for the hydraulic behaviour evaluation and to a module for water demand prediction using neural networks. The main objectives were the minimisation of operational costs and maximisation of benefits from energy generation. The study of Gonçalves and Ramos (2008) showed possible payback periods from 5 to 8 years for the installation of PATs.

The work of Ramos et al. (2010) offers a comparative study for pressure control by using a pressure reducing valve (PRV) and a PAT for simultaneous energy recovery. Similarly to Gonçalves and Ramos (2008), Ramos et al. (2010) stated that, for PATs in the range of 5 to 500 kW, the payback

time is significantly inferior than for a conventional turbine (less than 5 years). In their work, the authors developed a mathematical model linking EPANET to a GA for the minimisation of the nodal pressure by valve control subject to a constraint for the established minimum pressure. Performance curves of different pumps in the two operating modes (pump and turbine) were developed in order to select the best characteristic curve of the turbo-machine to be used to control pressure. The micro-hydro implementation was conducted in a case study at Algarve (Portugal), in a network beginning at the Beliche dam, passing through a pump station and continuing to the treatment plant. Results have shown similar operation conditions either using a PAT or a PRV. However, installations with hydropower systems allow a considerable energy reduction by using its own generation (Ramos et al., 2010).

A recent study dealing with quarter-hourly operation was published (J. Wang & Liu, 2011). The model presented in this study handles hydropower reservoirs with pump storage plants. The optimisation problem was divided into seven sub-horizons of 6 hours each and a quarter-hourly scheduling was performed. The authors believe that this procedure could be one of the keys to make this kind of problem easier to solve, avoiding, for example, the problem of the schedules providing simultaneous generating and pumping (identified in some 1-hour time-step problems according to J. Wang and Liu (2011)).

The work of Castronuovo and Lopes (2004) deals with the operational optimisation of a network containing a hybrid wind-hydropower plant. The wind power forecast is obtained by time-series, for a time horizon of 48 hours. The daily operation strategy for each one of the determined time-series scenario is optimised by a linear hourly-discrete algorithm. The predicted average economic gain of this optimisation was assumed to be between 425.3 and 716.9 k€ for an analysed test case. The authors also refer that the use of water storage ability allows improving the hybrid wind-hydro park profits. This is because the energy can be delivered to the network during the peak energy price (Castronuovo & Lopes, 2004).

Vieira and Ramos (2009) presented a work dealing with hydroturbines and wind power generation for pumping supply. A simplified system containing a pump station and excess of available energy in the gravity branch was tested. In order to use this excess of energy, the existent pressure reducing valve was replaced by a water turbine. A model based on LP and linked to EPANET was used to optimise the pump station operations during 24 hours of simulation and considering the reservoirs levels variations as the decision variables (Vieira & Ramos, 2009). Operations for both winter and summer conditions were also considered. With the pumping operations optimisation, energy costs were saved up to 47 %. Considering the inclusion of a wind turbine, the electricity needs for pumping the water were almost filled. The daily economic benefits obtained in the gravity system demonstrated to be dependent on the available energy, initial level of the downstream reservoir and its capacity (Vieira & Ramos, 2009).

Later, in Madeira (Portugal), the Socorridos WSS, a system with water consumption and inlet discharge, was tested with the methodology proposed by Vieira and Ramos (2008). The authors used Linear Programming and Non-linear Programming tools for the operational optimisation and again EPANET for the hydraulic simulation. Savings of nearly 100 €/day were obtained with the Non-linear Programming approach and, when a wind park is added to the system, the profits are approximately 5200 €/day. The pump and turbine controls, using NLP with and without wind turbines, were different from each other due to the non-linearity of the objective function and also due to the

availability of wind energy (Vieira & Ramos, 2008).

5.3 Water demand forecasting

Future water demand have been predicted for a variety of purposes, such as understand spatial and temporal patterns of water use, optimise system operations, plan for future system expansion or even prepare for future revenue and expenditures. According to the purpose of the forecast, distinct scales for the forecasting methods are defined, from short-term to long-term scales.

5.3.1 Time scales of forecasting models

Medium- to long-term forecasts (months to decades) are mostly used in strategic planning and to determine future resource requirements (Hyndman & Athanasopoulos, 2013). Sizing system capacity, staging system improvements and assessing how future environmental and economic conditions are likely to change water supply and demand are the main purposes of this forecast scale.

There are a number of recent works dealing with monthly water demand forecasting, applying from ARIMA models (Ghiassi, Zimbra, & Saidane, 2008) to ANN models (Babel & Shinde, 2011; Ghiassi et al., 2008) and also hybrid ANN models (Tiwari & Adamowski, 2013, 2014), where the ANN-based models commonly outperform the classic ones.

Short-term scales, in turn, are mostly used in scheduling processes (Hyndman & Athanasopoulos, 2013), including optimisation and management of systems operations. Hourly, daily and weekly forecasts are commonly included in this scale. Comparatively to the medium/long-term forecasts, a lot more studies can be found related to (i) hourly forecasts (Alvisi, Franchini, & Marinelli, 2007; Ghiassi et al., 2008; Herrera, Torgo, Izquierdo, & Pérez-García, 2010; H.-S. Kang et al., 2015; Martinez et al., 2007; Odan & Reis, 2012; Romano & Kapelan, 2014; Salomons et al., 2007; Santos & Pereira Filho, 2014), (ii) daily forecasts (J. Adamowski, Fung Chan, Prasher, Ozga-Zielinski, & Sliusarieva, 2012; J. F. Adamowski, 2008; Alvisi et al., 2007; Babel & Shinde, 2011; Bakker, Van Duist, Van Schagen, Vreeburg, & Rietveld, 2014; Ghiassi et al., 2008; Msiza, Nelwamondo, & Marwala, 2008; Tabesh & Dini, 2009) and (iii) weekly forecasts (J. Adamowski & Karapataki, 2010; K. Adamowski, Adamowski, Seidou, & Ozga-Zieliński, 2014; Bougadis, Adamowski, & Diduch, 2005; Ghiassi et al., 2008; Jain, Varshney, & Joshi, 2001; Tiwari & Adamowski, 2013, 2014).

Since this work is focused on short-term forecasting, more specifically in hourly scales, the literature review provided is focused in short-term water demand forecasting studies.

A resume on the analysis of the reviewed works are provided in appendix B. Figures B.1 and B.2 provide the results of the performance of the best distinct forecasting models obtained in each work. Figure B.3 resumes the input variables tested in each work as well as the ones that provided the best result for each case.

5.3.2 Short-term forecasting models

One of the first works applying artificial intelligence-based methods for the short-term water demand forecasting was the work of Jain et al. (2001). Since then, several authors have compared distinct innovative intelligence-based techniques with the traditional time series and regression methods. Jain et al. (2001) compare distinct Artificial Neural Networks (ANN) with the traditional methods for

the weekly water demand forecasting in an Indian Institute. The authors also analyse the influence of weather variables in this particular case-study, namely the maximum air temperature, the occurrence of rainfall and the rainfall amount. For this case-study, the ANN methods outperformed the autoregressive models (AR), the Multiple Linear Regression (MLR) and the Multiple Non-Linear Regression (MNLR). The authors also realised that, for both linear regression and ANN methods, the use of rainfall occurrence variables (binary) provides better results than considering the total rainfall amount.

Later, Bougadis et al. (2005) compared similar methods, including MLR, ARMA and ANN methods, for the peak weekly water demand forecasting in the city of Otawa (Canada), with the aim of developing an expansion strategy. The authors also tested the same input variables as Jain et al. (2001). Also in this case-study, the ANN outperformed the regression and time series models. However, concerning the input variables, the rainfall amount provided larger improvements in the models performance when compared to the rainfall occurrence (in opposition to the results of Jain et al. (2001)). At a first glance, these diverging results could be associated to the fact of two distinct case-studies (in different regions) are analysed.

Another work using a case-study also from the city of Otawa in Canada, however for the peak daily summer water demand forecasting, was published by J. F. Adamowski (2008). The results provided in this work demonstrate that the rainfall occurrence has a larger influence in the forecasting models performance. The main differences that stands out between this work and the work of Bougadis et al. (2005) is the time scale of the forecasting models (one weekly and other daily) and also the amount of data used for the forecasting models development. While Bougadis et al. (2005) used 4 months of data (around 16 weeks, 16 data points) to develop their models, J. F. Adamowski (2008) used 10 years of data, considering only the months from May to August (around 1200 days/data points). This means that the most adequate input variables for a certain forecasting model may be not only dependent on the model scale or time scale but also dependent on the amount of data available.

The work of J. F. Adamowski (2008) demonstrated that the ANN models are better predicting the peak daily demand than the regression or time series models.

In order to address the problem of drought in Cyprus, J. Adamowski and Karapataki (2010) developed a work to predict the peak weekly water demand in two different regions of Nicosia city (Athalassa and Public Garden). The authors decided to test a MLR model with ANN models differing in the training method: (i) Levenberg-Marquardt (LM), (ii) Conjugate Gradient Powell-Beale (CGPB) and (iii) Resilient Back-propagation. For both regions, the ANN-LM models provided average better results than all the other methods. Also the occurrence of rainfall demonstrated to be more relevant than the amount of rainfall, supporting the findings of Jain et al. (2001). In this case, 6 years of data (288 weeks) were used to develop the forecasting models.

The use of bootstrap data re-sampling techniques and wavelet analysis for the series decomposition applied to ANN models for water demand forecasting was introduced by Tiwari and Adamowski (2013). The authors tested such techniques for daily and weekly water demand forecast for the city of Montreal (Canada). Four distinct ANN methods were developed and compared to ARIMA models: (i) a simple ANN, (ii) a bootstrap artificial neural network (BANN), using bootstrap data samples from 100 ANN outputs, (iii) a wavelet ANN (WANN), using, as input, 4 distinct components of the series (water demand, temperature and precipitation) and (iv) a wavelet bootstrap ANN (WBANN), using 100 data samples of the wavelet series components (100 WANN). Results demonstrated that

the WANN outperform the other methods in the daily forecast, however, the WBANN provided better results in the weekly forecast scale.

Later, Tiwari and Adamowski (2014) and K. Adamowski et al. (2014) tested the previously mentioned bootstrap and wavelet-based ANN models for weekly water demand forecasting in the city of Calgary (Canada) considering limited data availability (around 2 years and 9 months). Similar to the previous case-study, the WBANN provided the best results, with the wavelet analysis improving the model performance and the bootstrap technique increasing the reliability of forecasts by producing ensemble forecasts.

Ghiassi et al. (2008) introduced a dynamic artificial neural network model (dynamic ANN) for weekly, daily and hourly water demand forecast and tested it in a case-study including the city of San Jose (California) and some other surrounding cities. The authors compare such developed forecasting method with a simple ANN and a traditional ARIMA and also investigate the influence of weather and anthropic variables (average temperature, type of day - week/weekend, and period of the day - day/night) in some models.

This dynamic ANN proposed by Ghiassi et al. (2008) defines the ANN architecture dynamically according to the complexity of the process and the desired accuracy. The outputs are obtained by a linear combination of three types of neurons: (i) constant elements, (ii) residuals non-linear elements and (iii) accumulated knowledge elements. At a first stage, the linear component is assessed and if the required accuracy is not reached, the model generates additional layers to capture the non-linear component of the process (by minimising the error).

For all time horizons (weekly, daily and hourly), the dynamic ANN proved to be effective achieving accuracies superior to the forecasting methods compared (Ghiassi et al., 2008).

In order to evaluate the effect of other variables, Ghiassi et al. (2008) decided to develop distinct models. For the weekly forecasts, after identifying distinct seasonal behaviour in data, the authors decided to divide the data into two sets (warm and cold seasons, *i.e.* May to September and October to April) and generate separated seasonal forecasts to be further combined and compared to the other model. This method improved the forecast accuracy by only 2 %.

Concerning the daily forecasting models, the authors decided to develop distinct models for week and weekend days (Monday to Friday and Friday to Sunday, where Friday is the transitional day). Forecast accuracy was improved by 22%, demonstrating the significant effect of these factors in this type of models (Ghiassi et al., 2008).

In the case of hourly forecasts, two distinct variables were tested for each representative month (April, the most difficult to predict and September, the most typical): (i) the type of day (week/weekend) and (ii) the period of the day (day/night). Thus, similarly to the weekly forecasts, the data was divided into two sets (week/weekend) to generate distinct models, which improved the forecasts accuracy by 8%. The other models were obtained by separating the data into day and night sets (5 a.m. to 2 p.m. and 3 p.m. to 4 a.m., respectively) and, in this case, forecast accuracy improvements of 9 % were obtained for the month of April and 14 % for the month of September. These last models for each period of the day were also tested using historical temperature variables as input. The forecast accuracy for the April model was improved by 8 % and for the September model by 18 %. This was expected since the temperature variation in September, for this case-study, is notably higher than in April.

Later, Odan and Reis (2012) applied the same dynamic ANN proposed by Ghiassi et al. (2008) in Araraqua city (Brazil) for the hourly water demand forecasting. The authors compared such method

with a simple ANN (testing distinct numbers of hidden layers) and also with hybrid ANN and hybrid dynamic ANN, consisting in the additional use of Fourier Series as input of the networks. The best results (both for 1h and 24h time horizon) were obtained with the dynamic ANN model using past demand observations and Fourier Series as input (not requiring weather information).

Babel and Shinde (2011) evaluated the effect of weather variables as ANN inputs for daily and monthly water demand forecast in the city of Bangkok (Thailand). In the daily forecasts, no significant differences were found in the forecast accuracy when including weather variables (rainfall, average temperature and relative humidity) in their models. On the other hand, in the monthly forecasts, the influence demonstrated to be slightly superior. However, in this last case other variables were additionally included in the models: population, per capita Gross Provincial Product, education status and household connections.

To face the problem of water scarcity in the South Africa's Gauteng Province, Msiza et al. (2008) developed a work on the daily water demand forecast using ANN and Support Vector Machines (SVM). The authors tested distinct training algorithms and distinct activation functions for Multilayer Perceptron and Radial Basis Function ANNs (ANN-MLP and ANN-RBF) and, in the case of the SVM, distinct kernel functions were also tested. The best results were obtained with an ANN-RBF using a linear activation function and a scaled conjugate gradient algorithm for training the model. This ANN-RBF model also outperformed the best SVM model.

Fuzzy and Neural-fuzzy forecasting techniques for daily water demand forecasting were proposed by Tabesh and Dini (2009) and compared with ANN using a case-study in Tehran (Iran). In order to face an expected water crisis, a short-term forecast of water demand in this city is necessary for the development of a water conservation programme to achieve the optimal operation of the networks. The authors found that fuzzy models, in general, do not produce good results for this case-study. However, the Neural-fuzzy models reveal to be comparable to ANN models, with similar forecast accuracy.

In the work of Tabesh and Dini (2009), the use of random input variables was also tested, demonstrating, in general, slightly improvements in the neural-fuzzy models performance. The best results were obtained with the models considering only past water demand variables as input (for both ANN and neural-fuzzy approaches).

Demand forecasting techniques have already been applied to real projects to support real-time optimal control of water supply systems. The research project called POWADIMA (POtable WAter DIstribution MAnagement) is an example of that (Alvisi et al., 2007; Martinez et al., 2007; Salomons et al., 2007). The idea of this project is to base the daily operational decisions on the expected future demands for water. Thus, a pattern-based methodology to forecast one-day ahead hourly water demand was developed. This specific methodology works based on two stages. The first stage uses Fourier Series to determine the average daily water demand taking into account the seasonal and weekly patterns. In the second stage, the determined daily term (periodic component) and a daily persistence component (based on the deviation computed using an AR process) is combined with the daily patterns and the hourly forecasts are determined using Time Series analysis. The hourly forecast deviation (what the authors call hourly persistence component) is hourly updated through a regression method whose coefficients are dependent on the hour of the day (Alvisi et al., 2007). The authors realise that updating the model with daily and hourly persistence components allow to improve the forecast performance. Such methodology was firstly tested in the municipality of

Castelfranco Emiliana in Italy (Alvisi et al., 2007) and then in larger case-studies such as the Haifa-A WSS in Israel (Salomons et al., 2007) and the WSS of the city of Valencia in Spain (Martinez et al., 2007). Similar forecast accuracy where obtained when comparing the smaller with the largest case-studies, which demonstrated the aptitude of the developed forecasting model for real applications.

Herrera et al. (2010) published an interesting work on the hourly water demand forecasting. These authors not only compare distinct non-linear forecasting methods but also provide a study on the data update for training the models. The idea of this study is to verify if the models demonstrate the same performance when using only the last observations (slide window approach, as denominated in the paper) as when using accumulated data (growing windows approach).

In the work of Herrera et al. (2010) the forecasting methods tested are: (i) simple 1 hidden layer ANN, (ii) Projection Pursuit Regression (PPR), a regression method that explains the target variable as a sum of spline functions of projections of the input variables, (iii) Multivariate Adaptive Regression Splines (MARS) that uses a linear combination of the splines to model the target variables, (iv) Support Vector Regression (SVR) that performs a non-linear mapping of data and obtains a linear regression model in the new space, (v) Random forests, an ensemble of tree-based regression models and (vi) weighted pattern-based model that uses the water demand seasonal properties (similar to the model proposed by Alvisi et al. (2007)). This last model was specifically used for comparison with the others. All the methods significantly outperformed the pattern-based method. The authors mention a disappointing performance of the ANN model, however, it should be notice that, in their study, no different number of hidden layers were tested, which was demonstrated in previous studies that may have a significant influence. Although all the models performance accuracy were quite similar, the SVR stand out.

Candelieri and Archetti (2014) also decided to use a SVM model for the hourly water demand forecast in the city of Milano (Italy), one of the case-studies of the ICeWater project (ICT Solutions for efficient Water Resources Management). The main difference introduced in this recent work applying SVM is the use of clusters that characterise the water demand patterns and the posterior training of separate SVM models for each distinct cluster identified in the raw data. The clusters are defined in daily time windows according to the time series similarity in time. For the tested case-study, six typical daily patterns (and thus, six clusters) were identified. The values of MAPE measured for the forecast with each cluster presented a 0.79%-14.33% range, with an average of 5.29%.

Recently, Santos and Pereira Filho (2014) published a work on the hourly water demand forecast in São Paulo Metropolitan area (Brazil). The authors decided to investigate the influence of distinct input variables (demand, anthropic and meteorological) with several lag times (up to 24 hours in the past) and distinct output lead times (forecast of hour 1, 6, 12, 18 and 24) in ANN models. The forecasting performance is also compared with a MLR model, which demonstrated to be less accurate than the best ANN. The only drawback observed in this study is that no ANN model is tested using only previous water demand (excluding anthropic and/or weather variables). At the same time, it is not clear what are the variables included in the MLR model. Since some of the ANN models performed worst then the classic MLR model, using different variables to develop the MLR model could also provide distinct results.

Romano and Kapelan (2014) test hourly ANN forecasting models (1 and 24 hours predicted in advance) using data measured at distinct zones in the Yorkhire WSS (United Kingdom): at 3 distinct District Metered Areas (DMA), each one supplying different population sizes, and at 1 reservoir

outlet. According to the provided results (see table B.2), no significant differences in the forecast accuracy are found for the distinct model scales tested (comparing MAPE and NSE accuracy measures). The authors tested the use of an Evolutionary Algorithm to automatically find the best parameters and structure of the ANN (EA-ANN) instead of using fixed user-defined structures. This approach provided significant improvements in the models performance (Romano & Kapelan, 2014).

Another approach proposed by Romano and Kapelan (2014) for the 24h forecast horizon is the use of multiple parallel ANN (for each hour to predict) instead of a single ANN. Results illustrated slightly improvements (but with increased computational effort) using such approach for both the EA-ANN and the fixed-structure ANN.

For the optimal control and to detect pipe bursts in water distribution networks, Bakker (2014) (also (Bakker, Vreeburg, Van Schagen, & Rietveld, 2013)) proposed the use of 15-min time-steps to better describe the water demand variations instead of the typical 1-hour time-steps. Since the existent models to detect pipe bursts are typically in smaller time-steps, Bakker (2014) decided to use such time scales for water demand forecasting and pumps control. This approach was used to predict the future 48 hours of water demand in six different cities in the Netherlands (all with differing number of consumers). In order to implement the forecasting model in real WSS, the authors developed a pattern-based model that only uses past water demand and anthropic variables (the seven days of the week and other specific deviating day types) and uses a functionality based on factors update to identify deviating weather-related water demand. Thus, at each 15-min time-step, a new 48h forecast is calculated considering the updated factors. To assess the developed forecast model, the authors computed the accuracy measures considering a 24-hour forecast intervals (obtained from the average of the 15-min steps) and 15-min intervals. Although the RMSE and the MAPE presented significantly better (lower) values for the 24-hour forecast, the 15-min forecast demonstrated a better fit, providing higher values for the NSE. The case-studies of cities with a larger number of consumers also demonstrate to be easier to predict Bakker (2014); Bakker et al. (2013), which is showing again the dependence of the forecasting models with the model scales.

The previously mentioned case-studies from the Netherlands were also used to test the performance of forecasting models using weather variables (average daily temperature, in this case) as input Bakker (2014); Bakker et al. (2014). Daily forecasts were computed using the same datasets. Results obtained using the previously described adaptive pattern-based method were compared with a MLR model and a transfer-/noise method (combination of an ARIMA model with a linear transfer model).

Using the weather variables, the largest forecasting errors were reduced by 9.4 % and the average by 6.3 % (Bakker, 2014; Bakker et al., 2014) even in these case-studies with low variability in weather conditions, which means that for other case-studies presenting higher weather influences, the differences can be larger. Concerning the distinct methods applied, although the introduced tranfer-/noise model provided slightly better results than the adaptive pattern-based model, the authors mention that the later may be better accepted for real implementations since it is easier to understand by the control operators.

In other recent works for the hourly water demand forecast, instead of applying the innovative machine learning techniques that demonstrated good performances in previous works, researchers have decided to use combinations of classic ARIMA and Exponential Smoothing methods in order to improve the models performance (such as Y. Wang, Ocampo-Martinez, Puig, and Quevedo (2014)

and H.-S. Kang et al. (2015)).

Y. Wang et al. (2014) proposed a Double-Seasonal multiplicative Holt-Winters model combined with a Gaussian Process regression (with uncertainty propagation) for multiple-step ahead forecasts (since a typical exponential smoothing forecast is only based on the last known value). This approach was applied to the WSS of Barcelona (Spain).

(H.-S. Kang et al., 2015) combined an ARIMA model with Exponential Smoothing to forecast the hourly water demand in a WSS in the Gallella region (Sri Lanka), typically a rural area. While the ARIMA-alone method failed to predict the lower water demands, the combination with the Exponential Smoothing method with the smoothing parameter α set to 0.9 allowed to overcome such drawback, improving the forecast accuracy. Although this results are not compared with other methods, it is possible to compare this case-study (with an average demand of 450 m^3/day) with another rural area in Netherlands, the city of Hulsber (440 m^3/day) presented by Bakker (2014). Comparing the values of the accuracy measures (see table B.2), it is observed that the models used by Bakker (2014) provided more accurate results.

From the analysis to the previously mentioned works, it is possible to understand that, the data analysis and pre-processing represents an important role in the forecasting process with influence in the models accuracy.

Although distinct amounts of data have been used to develop the distinct forecasting models, the work of (Herrera et al., 2010) demonstrated that, for hourly forecasts, the accumulation of available data for training the models does not provide significant changes in the models performance. This means that using the most recent weeks of available data should be enough to train hourly water demand forecasting models. The only problem on following this approach is the possible occurrence of failures in the data measurement and/or communication, which can significantly reduce the amount of existent data for training the model. Thus, it is recommended to always select a larger amount of data (if possible).

The mean absolute relative errors for the best hourly forecast models reviewed in this work are typically in the 3.4 % – 15.0 % range for ARIMA and regression models and in the 0.8 % – 14.3 % range for models based on artificial intelligence.

It is also possible to conclude that to develop an automatic forecasting model for operation in real-time, two main characteristics should be presented by the model: (i) scalability, to provide good performances for distinct model scales, and (ii) adaptability, to easily adapt to possible changes in the WSS conditions (self-learning capability) and avoid unnecessary model maintenance and calibration (and thus, avoiding the associated cots).

6. Overview and discussion

The main identified obstacles to the introduction of works developed in academia into industrial environment are discussed. Some tips to overcome such obstacles are provided. A general discussion on the addressed topics and some relevant conclusions are presented.

6.1 Water industry requirements

Most developed methodologies for water systems improvement have been oriented towards determining least-cost design and pump-scheduling strategies. However, the acceptance of some innovative methods by water industries for real applications can be partially limited due to (Rao & Salomons, 2007): (a) the confinement of some techniques to minimise energy costs ignoring the network performance, (b) the complexity of the problem formulation due to considerable amount of mathematical sophistication, (c) the complexity of the networks that is dependent of their size, (d) usually oversimplification of the systems and (e) the excessive run times and easy trapping at local optima.

For a better acceptance of the methods by the industry, it is important to develop robust software programmes applying the corresponding methods with: (a) intuitive and attractive graphic interfaces, (b) easy adaptation to new situations and, maybe the most important issue, (c) a specific attention to the network performance and the consumers supply requirements. For the guarantee of this last topic, the use of a calibrated hydraulic simulator in order to reflect the true operational characteristics of the networks is crucial. The calibration process requires a considerable amount of data, usually collected manually, and manipulation of model variables (Machell et al., 2010). Bunn (2007) also pointed out that any optimised solution must (i) be obtained quickly enough to respond to real-time changes in the system, (ii) not interfere with protection mechanisms and (iii) not cause negative impact on water quality.

The best way to reach all this objectives is working in parallel with water industries during the methodologies development in order to respond to particular industry requirements. At the same time it is important to notice that the best way to meet all the needs implies the use of a combination of complementary tools in order to manage the complexity of water-related challenges (Marton-Lefevre & Bakker, 2012).

For the analysis of real water systems efficiency, global and local water situations should be assessed and then a focus on critical points should be done, implementing action and setting targets. During these steps, water risks must be identified and controlled through monitor and communication

means. Even with all the steps completed, water companies should revisit their strategies and reassess opportunities for continuous improvement (Marton-Lefevre & Bakker, 2012).

6.2 General discussion

Water supply systems are characterised by a large variety and complexity providing a huge opportunity to act in the economic and/or energetic efficiency improvement. For the distinct existing networks there are several possible measures to apply, where some of them imply large investment and others not so much.

Although the installation of variable-speed drives in pumps allied to the control optimisation may provide the greatest reductions of energy and costs associated to the network operation, it should be noticed that this measure is more capable of providing such results in installations presenting flow rate variation. For installations with no flow rate variation, the energy tariff can always be checked and maybe changed by a more suitable one.

Despite the high investment cost, the implementation of hydroturbines in areas containing excessive pressures can be an attractive measure when presenting low payback times.

The use of hydraulic simulators is essential during the application of improvement measures in order to guarantee proper operation of the networks. The choice of the hydraulic simulator presents a significant impact on the optimisation process since it determines the feasibility of the optimal solutions. A quick evidence of this fact can be observed, for instance, when some solutions obtained in older studies are considered not feasible in current studies due the use of of improved hydraulic simulators. It is observed that studies making use of EPANET 2.0, for example, have resulted in better solutions (see resume of results in Appendix A).

For real applications, the use of calibrated hydraulic models is decisive for the success of the improvement measures. The process of calibration of such models can usually represent the most time-consuming step in the entire optimisation process of a WSS.

Still concerning the optimisation, it is possible to identify the need of some studies comparing the performance of several algorithms in a networks containing all possible complexities existent in the real world. In the case of the design optimisation, the use of benchmark networks for comparative studies has been very common. However, the same is not occurring for operational optimisation. Another important issue is related to the lack of comparative studies performed under the same conditions such as the constraints and initial parameters considered for the networks modelling.

One of the main questions that is pointed out for the control optimisation problems is concerning the decision for the optimisation variables formulation. It is not clear that an explicit formulation is the best choice since it deals directly with pumps. At the same time, it is also not clear that the alternative implicit formulation (such as the tanks water levels variation) provide better solutions. When dealing with fixed-speed pumps and, at the same time, when tanks are directly connected to the pump stations, the relation between tanks water levels variations and the desired pump controls for such variations can be easy. However, in complex networks, the decision of the best pumps operation according to certain variations in water tanks levels can be difficult. It is the case when the networks present variable-speed pumps, several pumps with distinct characteristics, multiple storage tanks with no direct connexion to the pump stations, etc. In such cases, the explicit optimisation of pumps (and even valves) may allow achieving better solutions. On the other hand, dynamic optimisation could be

an option to solve these more complex cases through an implicit formulation.

Regarding the influence of the connexion between real-time systems and optimisation modules, it is worth noticing that no investigations were made in the ability of the algorithms to adapt to small changes in constraints conditions such as online changes in predicted demand and current reservoir levels.

By analysing the collected works related to water demand prediction, the advantage of combine distinct forecasting models becomes clear. Artificial Neural Networks, for example, can present a great performance when enough historical data is available for the process of training required by this method. However, when these kinds of methods fail, time-series and regression models can provide some guarantees. It is most important to refer that the stage of water prediction during an optimisation process determines the success of reaching the best solution.

Scientific works, as the ones discussed in this part, developed in parallel with water companies, present higher probability of success and feasibility in the large variety of real systems since they allow important adaptations during the development and the implementation of innovative techniques and methodologies. Furthermore, as it became clear in this discussion, both the reviewed subjects contribute decisively for the achievement of the best efficiency improvements and, for this reason, should not be treated separately. The process of optimisation of a water network, for example, can never provide significant improvements (i) if the hydraulic model for simulation is not adequate, (ii) if the water demand prediction model fails, or even (iv) if the formulation of the problem being solved does not consider specific requirements of the system.

Since the attention to the sustainability in the world is continuously growing, the importance of including hydro, wind and/or solar energy systems in this field is also increasing. This inclusion not only contributes towards the sustainability of the water systems but also promotes the development of clean and renewable energy production, reducing the dependence of fossil fuels and consequently reducing environmental impact. However, as already referred by Ramos et al. (2010), some energy policies required to impose the implementation of these strategies by water companies are missing.

The concept of smart water networks (or smart grids for water) is an emerging market similar and parallel to the established electrical smart grids. According to the IDC Energy Insights report “Smart Water Market Overview” (SmartGridNews, 2012), smart water network management solutions will grow faster than smart water metering and the major benefits will correspond to water loss reduction.

The solutions presented in this state-of-the-art review, especially the ones related to the networks automation and real-time operation, can reveal a huge contribution for the development of this recent concept.

References

Abebe, A., & Solomatine, D. (1998). Application of global optimization to the design of pipe networks. In *Proceedings of the 3rd international conference on hydroinformatics* (Vol. 98, pp. 989–995).

Adamowski, J., Fung Chan, H., Prasher, S. O., Ozga-Zielinski, B., & Sliusarieva, A. (2012). Comparison of multiple linear and nonlinear regression, autoregressive integrated moving average, artificial neural network, and wavelet artificial neural network methods for urban water demand forecasting in montreal, canada. *Water Resources Research*, *48*(1).

Adamowski, J., & Karapataki, C. (2010). Comparison of multivariate regression and artificial neural networks for peak urban water-demand forecasting: evaluation of different ann learning algorithms. *Journal of Hydrologic Engineering*, *15*(10), 729–743.

Adamowski, J. F. (2008). Peak daily water demand forecast modeling using artificial neural networks. *Journal of Water Resources Planning and Management*, *134*(2), 119–128.

Adamowski, K., Adamowski, J. F., Seidou, O., & Ozga-Zieliński, B. (2014). Weekly urban water demand forecasting using a hybrid wavelet–bootstrap–artificial neural network approach. *Annals of Warsaw University of Life Sciences, Land Reclamation*, *46*(3), 197–204.

Alperovits, E., & Shamir, U. (1977). Design of optimal water distribution systems. *Water resources research*, *13*(6), 885–900.

Alvisi, S., Franchini, M., & Marinelli, A. (2007). A short-term, pattern-based model for water-demand forecasting. *Journal of Hydroinformatics*, *9*(1), 39–50.

Amit, R., & Ramachandran, P. (2009). *Optimal design of water distribution networks a review.* `http://works.bepress.com/rkamit/7/`.

Aquis: 7-technologies. (2012). `http://www.7t.dk/products/aquis/product-information/aquis-operation.aspx`. (Accessed: 2012)

Atkinson, R., Van Zyl, J., Walters, G., & Savic, D. (2000). Genetic algorithm optimisation of level-controlled pumping station operation. *Water network modelling for optimal design and management*, 79–90.

Babel, M. S., & Shinde, V. R. (2011). Identifying prominent explanatory variables for water demand prediction using artificial neural networks: a case study of bangkok. *Water resources management*, *25*(6), 1653–1676.

Bakker, M. (2014). *Optimised control and pipe burst detection by water demand forecasting* (Unpublished doctoral dissertation). TU Delft, Delft University of Technology.

Bakker, M., Van Duist, H., Van Schagen, K., Vreeburg, J., & Rietveld, L. (2014). Improving the performance of water demand forecasting models by using weather input. *Procedia Engineering*, *70*, 93–102.

Bakker, M., Vreeburg, J., Van Schagen, K., & Rietveld, L. (2013). A fully adaptive forecasting model for short-term drinking water demand. *Environmental Modelling & Software*, *48*, 141–151.

Barnett, M., Lee, T., Jentgen, L., Conrad, S., Kidder, H., Woolschlager, J., ... Hollifield, D. (2004). Real-time automation of water supply and distribution for the city of jacksonville, florida, usa. *Water Supply*, 15–29.

Bene, J. G., & Hős, C. J. (2011). Finding least-cost pump schedules for reservoir filling with a variable speed pump. *Journal of Water Resources Planning and Management*, *138*(6), 682–686.

Bentley: Watercad v8i. (2010). `http://www.bentley.com/en-US/Products/WaterCAD/`. (Accessed: 2012)

Bougadis, J., Adamowski, K., & Diduch, R. (2005). Short-term municipal water demand forecasting. *Hydrological Processes*, *19*(1), 137–148.

Bragalli, C., D'Ambrosio, C., Lee, J., Lodi, A., & Toth, P. (2012). On the optimal design of water distribution networks: a practical minlp approach. *Optimization and Engineering*, *13*(2), 219–246.

Brion, L. M., & Mays, L. W. (1991). Methodology for optimal operation of pumping stations in water distribution systems. *Journal of Hydraulic Engineering*, *117*(11), 1551–1569.

Bullnheimer, B., Hartl, R. F., & Strauss, C. (1997). *A new rank based version of the ant system. a computational study.* SFB Adaptive Information Systems and Modelling in Economics and Management Science, WU Vienna University of Economics and Business.

Bunn, S. (2007). Closing the loop in water supply optimisation. In *Proceedings of the 2007 iet water event* (pp. 71–82).

Candelieri, A., & Archetti, F. (2014). Identifying typical urban water demand patterns for a reliable short-term forecasting–the icewater project approach. *Procedia Engineering*, *89*, 1004–1012.

Carravetta, A., Del Giudice, G., Fecarotta, O., & Ramos, H. M. (2012). Energy production in water distribution networks: A pat design strategy. *Water resources management*, *26*(13), 3947–3959.

Carrijo, I. B., Reis, L. F. R., Walters, G. A., & Savic, D. (2004). Operational optimization of wds based on multiobjective genetic algorithms and operational extraction rules using data mining. In *Proceedings of the world water and environmental resources congress.*

Carroll, C. W. (1961). The created response surface technique for optimizing nonlinear, restrained systems. *Operations Research*, *9*(2), 169–184.

Caseiro, J., Valente, R., Andrade-Campos, A., & Yoon, J. (2011). Elasto-plastic buckling of integrally stiffened panels (isp): An optimization approach for the design of cross-section profiles. *Thin-Walled Structures*, *49*(7), 864–873.

Castronuovo, E. D., & Lopes, J. (2004). On the optimization of the daily operation of a wind-hydro power plant. *IEEE Transactions on Power Systems*, *19*(3), 1599–1606.

Caxaria, G. A., Sousa, D. d. M., & Ramos, H. M. (2011). Small scale hydropower: generator analysis and optimization for water supply systems. *Hydropower Applications*, *6*, 1386.

Cembrano, G., Brdyś, M., Quevedo, J., Coulbeck, B., & Orr, C. (1988). Optimization of a multi-reservoir water network using a conjugate gradient technique. a case study. In *Analysis and optimization of systems* (pp. 987–999). Springer.

Cembrano, G., Wells, G., Quevedo, J., Pérez, R., & Argelaguet, R. (2000). Optimal control of a water distribution network in a supervisory control system. *Control engineering practice*,

8(10), 1177–1188.

Cheung, P. B., Reis, L. F., Formiga, K. T., Chaudhry, F. H., & Ticona, W. G. (2003). Multiobjective evolutionary algorithms applied to the rehabilitation of a water distribution system: A comparative study. In *Evolutionary multi-criterion optimization* (pp. 662–676).

Coelho, B., & Andrade-Campos, A. (2012). Using different strategies for improving efficiency in water supply systems. In *Proceedings of the first eccomas young investigators conference on computational methods in applied sciences.*

Coelho, B., Tavares, A., & Andrade-Campos, A. (2012). Analysis of diverse optimisation algorithms for pump scheduling in water supply systems. In *Proceedings of engopt 2012 - 3rd international conference on engineering optimization.*

Coello, C. C., Lamont, G. B., & Van Veldhuizen, D. A. (2007). *Evolutionary algorithms for solving multi-objective problems*. Springer Science & Business Media.

Coin-or: Bonmin homepage. (2011). :`http://www.coin-or.org/Bonmin/`. (Accessed: 2012)

Costa, A. L. H., Medeiros, J. L., & Pessoa, F. L. P. (2000). Optimization of pipe networks including pumps by simulated annealing. *Brazilian Journal of Chemical Engineering*, *17*(4-7), 887–896.

Coulbeck, B., Orr, C.-H., & Brdyś, M. (1988). Real-time optimized control of water distribution systems. In *IET International Conference on Control* (pp. 634–640).

Cunha, A. A. R. (2009). *Otimização energética em tempo real da operação de sistemas de abastecimento de água.* (Unpublished master's thesis). Universidade de São Paulo.

Cunha, M. C., & Ribeiro, L. (2004). Tabu search algorithms for water network optimization. *European Journal of Operational Research*, *157*(3), 746–758.

Dandy, G. C., Simpson, A. R., & Murphy, L. J. (1996). An improved genetic algorithm for pipe network optimization. *Water Resources Research*, *32*(2), 449–458.

Deb, K., Agrawal, S., Pratap, A., & Meyarivan, T. (2000). A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: Nsga-ii. *Lecture notes in computer science*, *1917*, 849–858.

Derceto, inc: Introducing derceto aquadapt. (2011). :`http://www.derceto.com/Products-Services/Derceto-Aquadapt`. (Accessed: 2012)

Dhi: Mike - modelling the world of water. (2011). `http://www.mikebydhi.com/`. (Accessed: 2012)

Djebedjian, B., Herrick, A., & Rayan, M. A. (2000). Modeling and optimization of potable water network. In *International pipeline conference and technology exposition 2000* (pp. 1–5).

Doig, A. (2002). *Micro-hydro power: Practical action.* `http://practicalaction.org/media/download/10434`. The Schumacher Centre for Technology and Development, Warwickshire, UK. (Accessed: 2012)

Dorsch gruppe: Dc water design extension. (2006). `http://dcwaterdesign.sourceforge.net/`. (Accessed: 2012)

Dötsch, F., Denzinger, J., Kasinger, H., & Bauer, B. (2010). Decentralized real-time control of water distribution networks using self-organizing multi-agent systems. In *2010 4th IEEE International Conference on Self-Adaptive and Self-Organizing Systems (SASO)* (pp. 223–232).

Eiger, G., Shamir, U., & Ben-Tal, A. (1994). Optimal design of water distribution networks. *Water resources research*, *30*(9), 2637–2646.

Eleotero, B. C. (2008). *Reduction of electric energy costs in pumping systems - a case study in the*

water supply system of capinzal/ouro - sc (Unpublished master's thesis). Universidade Federal de Santa Catarina, Florianópolis, SC.

El Mouatasim, A., Ellaia, R., & Al-Hossain, A. (2012). A continuous approach to combinatorial optimization: application of water system pump operations. *Optimization Letters*, *6*(1), 177–198.

Engineering, H. E. (2008). *Hydrologis s.r.l.* `https://sites.google.com/a/hydrologis.com/hydrologis-environmental-engineering-en/`. (Accessed: 2012)

Europump and Hydraulic Institute. (2004). *Variable Speed Pumping – A Guide to Successful Applications*. Elsevier Advanced Technology. Retrieved from `https://www1.eere.energy.gov/manufacturing/tech_assistance/pdfs/variable_speed_pumping.pdf` (Executive Summary)

Fallside, F., & Perry, P. (1975). Hierarchical optimisation of a water-supply network. In *Proceedings of the institution of electrical engineers* (Vol. 122, pp. 202–208).

Farmani, R., Walters, G., & Savic, D. (2006). Evolutionary multi-objective optimization of the design and operation of water distribution network: total cost vs. reliability vs. water quality. *Journal of Hydroinformatics*, *8*(3), 165–179.

Feldman, M. (2009). Aspects of energy efficiency in water supply systems. In *Proceedings of the 5th iwa water loss reduction specialist conference, south africa* (pp. 85–89).

Fetyan, K. M., Younes, M., Helal, M., & Hallouda, M. M. (2007). Energy saving of adjustable speed pump stations in egypt. In *Proceedings of 11th International Water Technology Conference. Sharm El-Sheikh, Egypt: IWTC11.*

Firmino, M., Albuquerque, A., Curi, W., & Silva, N. (2006). Method for energy efficiency in water pumping by linear programming and integer linear programming. *Proceedings of VI SEREA – Seminário Iberoamericano sobre Sistemas de Abastecimento Urbano de Água. João Pessoa, Brazil*.

Fontana, N., Giugni, M., & Portolano, D. (2011). Losses reduction and energy production in water-distribution networks. *Journal of Water Resources Planning and Management*, *138*(3), 237–244.

Formiga, K., Chaudhry, F., & Vieria, M. (2006). Multi-objective optimization of water supply networks. *Proceedings of VI SEREA – Seminário Iberoamericano sobre Sistemas de Abastecimento Urbano de Água. João Pessoa, Brazil*.

Geem, Z. W. (2006). Optimal cost design of water distribution networks using harmony search. *Engineering Optimization*, *38*(03), 259–277.

Geem, Z. W. (2009). Harmony search optimisation to the pump-included water distribution network design. *Civil Engineering and Environmental Systems*, *26*(3), 211–221.

Gellings, C. W. (2009). Program on technology innovation: electric efficiency through water supply technologies - a roadmap. *EPRI, Electric Power Research Institute*.

Ghiassi, M., Zimbra, D. K., & Saidane, H. (2008). Urban water demand forecasting with a dynamic artificial neural network model. *Journal of Water Resources Planning and Management*, *134*(2), 138–146.

Gl water. (2012). `http://www.gl-group.com/en/water/SynerGEEWater.php`. (Accessed: 2012)

Goldman, F. E., & Mays, L. W. (1999). The application of simulated annealing to the optimal oper-

ation of water systems. In *29th annual water resources planning and management conference* (pp. 6–9).

Goldman, F. E., & Mays, L. W. (2005). *Water distribution system operation: Application of simulated annealing.* McGraw-Hill: New York, NY, USA.

Gomes, A., Albuquerque, C., Frangipani, M., de Combate ao Desperdício de Água, P. N., Gonçalves, E., de Modernização do Setor Saneamento (Brasil), P., ... de Paula Coura, S. (2007). *Guias práticos - técnicas de operação em sistemas de abastecimento de água: A conta de energia elétrica no saneamento*. Ministério das Cidades. Retrieved from `https://books.google.pt/books?id=ppKtNQAACAAJ`

Gonçalves, F. V., & Ramos, H. M. (2008). Controlo económico e energético e proposta de optimização. *Alterações Climáticas e Gestão da Água e Energia em Sistemas de Abastecimento e Drenagem*, 164–171.

Halcrow: Water supply and distribution systems optimization software. (2011). `http://www.halcrow.com/encoms`. (Accessed: 2012)

Halhal, D., Walters, G. A., Ouazar, D., & Savic, D. (1997). Water network rehabilitation with structured messy genetic algorithm. *Journal of Water Resources Planning and Management*, *123*(3), 137–146.

Helix technologies: Helix delta-q pipe networks. (2011). `http://www.helixtech.com.au/Q2Main.aspx`. (Accessed: 2012)

Herrera, M., Torgo, L., Izquierdo, J., & Pérez-García, R. (2010). Predictive models for forecasting hourly urban water demand. *Journal of hydrology*, *387*(1), 141–150.

Hunter gis: Epanet inp support module. (2012). `http://www.hunter-gis.com/Sol-EPANET.html`. (Accessed: 2012)

Hydraulicad. (1998). `http://www.hydraulicad.com/`. (Accessed: 2012)

Hyndman, R., & Athanasopoulos, G. (2013). *Forecasting: principles and practice*. Retrieved 2015, from `http://otexts.org/fpp/`

INAR. (2003). *Inar - technology solutions for the enterprise.* `http://www.inar.net/products/aquanet.htm`. (Accessed: 2012)

Ingeduld, P. (2007). Real-time forecasting with epanet. In *Proceedings World Environmental and Water Resources Congress: Restoring Our Natural Habitat. Tampa, Florida* (pp. 15–17).

Innovyze. (2012). *Innovyze products.* `http://www.innovyze.com/products/`. (Accessed: 2012)

Jain, A., Varshney, A. K., & Joshi, U. C. (2001). Short-term water demand forecast modelling at IIT Kanpur using artificial neural networks. *Water Resources Management*, *15*(5), 299–321.

James, K., Godlove, C. E., & Campbell, S. L. (2002). *Água e energia: Aproveitando as oportunidades de eficientização de água e energia não exploradas nos sistemas de água municipais.* Alliance - Aliança para Conservação de Energia.

Kang, D., & Lansey, K. (2011). Revisiting optimal water-distribution system design: issues and a heuristic hierarchical approach. *Journal of Water resources planning and management*.

Kang, H.-S., Kim, H., Lee, J., Lee, I., Kwak, B.-Y., & Im, H. (2015). Optimization of pumping schedule based on water demand forecasting using a combined model of autoregressive integrated moving average and exponential smoothing. *Water Science & Technology: Water Supply*, *15*(1), 188–195.

Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. In *1995 Proceedings of the IEEE*

International Conference on Neural Networks (Vol. 4, pp. 1942–1948).

Kessler, A., & Shamir, U. (1989). Analysis of the linear programming gradient method for optimal design of water supply networks. *Water Resources Research*, *25*(7), 1469–1480.

Kiselychnyk, O., Bodson, M., & Werner, H. (2009a). Overview of energy efficient control solutions for water supply systems. *Transactions of Kremenchuk State Polytechnic University*, *3*(56), 40–5.

Kiselychnyk, O., Bodson, M., & Werner, H. (2009b). Overview of energy efficient control solutions for water supply systems. *Transactions of Kremenchuk State Polytechnic University*, *3*(56), 40–5.

Kolhe, M., Joshi, J., & Kothari, D. (2004). Performance analysis of a directly coupled photovoltaic water-pumping system. *IEEE Transactions on Energy Conversion*, *19*(3), 613–618.

KYPipe. (2009). *Pipe2010: Kypipe overview.* `http://kypipe.com/kypipe`. (Accessed: 2012)

Liong, S.-Y., & Atiquzzaman, M. (2004). Optimal design of water distribution network using shuffled complex evolution. *Journal of The Institution of Engineers, Singapore*, *44*(1), 93–107.

Lopes, R. E., & MARTINEZ, B. (2006). Uso de bombas funcionando como turbinas para sistemas de recalque de água. *VI SEREA – Seminário Iberoamericano sobre Sistemas de Abastecimento Urbano de Água, João Pessoa*.

López-Ibáñez, M. (2009). *Operational optimisation of water distribution networks.* (Unpublished doctoral dissertation). Edinburgh Napier University.

López-Ibáñez, M., Prasad, D., & Paechter, B. (2008). Parallel optimisation of pump schedules with a thread-safe variant of epanet toolkit. In *Proceedings of the 10th Annual Water Distribution Systems Analysis Conference* (pp. 1–10).

López-Ibáñez, M., Prasad, T. D., & Paechter, B. (2005). Multi-objective optimisation of the pump scheduling problem using spea2. In *2005 IEEE Congress on Evolutionary Computation* (Vol. 1, pp. 435–442).

López-Ibáñez, M., Prasad, T. D., & Paechter, B. (2011). Representations and evolutionary operators for the scheduling of pump operations in water distribution networks. *Evolutionary computation*, *19*(3), 429–467.

Machell, J., Mounce, S., & Boxall, J. (2010). Online modelling of water distribution systems: a uk case study. *Drinking Water Engineering and Science*, *3*, 21–27.

Macke, S. (2012). *Epanet.de - hydraulic network analysis.* `http://epanet.de/documentation/index.html`. (Accessed: 2012)

Mackle, G., Savic, D., & Walters, G. A. (1995). Application of genetic algorithms to pump scheduling for water supply. In *First International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications. GALESIA. (No. 414)* (pp. 400–405).

Maier, H. R., Simpson, A. R., Zecchin, A. C., Foong, W. K., Phang, K. Y., Seah, H. Y., & Tan, C. L. (2003). Ant colony optimization for design of water distribution systems. *Journal of water resources planning and management*, *129*(3), 200–209.

Mallipeddi, R., & Suganthan, P. N. (2010). Ensemble of constraint handling techniques. *IEEE Transactions on Evolutionary Computation*, *14*(4), 561–579.

Martinez, F., Hernandez, V., Alonso, J., Rao, Z., & Alvisi, S. (2007). Optimizing the operation of the valencia water-distribution network. *Journal of Hydroinformatics*, *9*(1), 65–78.

Martins, V. A., Perez, A. M. E. A., Bardales, K. L. N., Gonçalves, F. V., Cheung, P. B., & Ide, C. N.

(2006). Modelagem computacional como ferramenta para estudos de eficiência energética no saneamento. *VI SEREA – Seminário Iberoamericano sobre Sistemas de Abastecimento Urbano de Água, João Pessoa*.

Marton-Lefevre, J., & Bakker, P. (2012). *Water for business - initiatives guiding sustainable water management in the private sector* (3rd ed.). `http://www.bcsd.org.tw/sites/default/files/node/domain_tool/678.file.2161.pdf`. World Business Council for Sustainable Development. (Accessed: 2012)

Mays, L. W. (1989). Reliability analysis of water distribution systems..

Michalewicz, Z. (1995). A survey of constraint handling techniques in evolutionary computation methods. *Evolutionary Programming*, *4*, 135–155.

Montalvo, I., Izquierdo, J., Pérez, R., & Tung, M. M. (2008). Particle swarm optimization applied to the design of water supply systems. *Computers & Mathematics with Applications*, *56*(3), 769–776.

Msiza, I. S., Nelwamondo, F. V., & Marwala, T. (2008). Water demand prediction using artificial neural networks and support vector regression. *Journal of computers*, *3*(11), 1–8.

Muljadi, E. (1997). Pv water pumping with a peak-power tracker using a simple six-step square-wave inverter. *IEEE Transactions on Industry Applications*, *33*(3), 714–721.

Nayar, C., Vasu, E., & Phillips, S. (1993). Optimised solar water pumping system based on an induction motor driven centrifugal pump. In *Proceedings of the IEEE 10th Conference on Computer, Communication, Control and Power Engineering* (Vol. 5, pp. 388–393).

NeatWork. (2010). *Neatwork - water distribution network.* `http://neatwork.ordecsys.com/index.html`. (Accessed: 2012)

Nelder, J. A., & Mead, R. (1965). A simplex method for function minimization. *The computer journal*, *7*(4), 308–313.

Odan, F. K., & Reis, L. F. R. (2012). Hybrid water demand forecasting model associating artificial neural network with fourier series. *Journal of Water Resources Planning and Management*, *138*(3), 245–256.

Olsson, R., Kapelan, Z., & Savic, D. (2009). Probabilistic building block identification for the optimal design and rehabilitation of water distribution systems. *Journal of Hydroinformatics*, *11*(2), 89–105.

Optimatics. (2011). *Optimizer wds software - water distribution systems.* `http://optimatics.com/software/optimizer-wds`. (Accessed: 2012)

OptiWater. (2012). *optidesigner.* `http://www.optiwater.com/optidesigner.html`. (Accessed: 2012)

Ormsbee, L., Lingireddy, S., & Chase, D. (2009). Optimal pump scheduling for water distribution systems. In *Multidisciplinary international conference on scheduling: Theory and applications (mista 2009)* (pp. 10–12).

Ostfeld, A., & Tubaltzev, A. (2008). Ant colony optimization for least-cost design and operation of pumping water distribution systems. *Journal of Water Resources Planning and Management*, *134*(2), 107–118.

Ouarda, T., & Labadie, J. (2001). Chance-constrained optimal control for multireservoir system optimization and risk analysis. *Stochastic environmental research and risk assessment*, *15*(3), 185–204.

Pegg, S. (2001). An online optimised pump scheduling system. In *Proceedings of the orsnz conference twenty naught one* (Vol. 170).

Perelman, L., Ostfeld, A., & Salomons, E. (2008). Cross entropy multiobjective optimization for water distribution systems design. *Water Resources Research*, *44*(9).

PROTEO S.p.A. (2012). *Products and services.* `http://www.proteo.it/prodotti/prodotti.asp`. (Accessed: 2012)

Ramos, H., Mello, M., & De, P. (2010). *Clean power in water supply systems as a sustainable solution - from planning to practical implementation*. IWA Publishing.

Rao, Z., & Salomons, E. (2007). Development of a real-time, near-optimal control process for water-distribution networks. *Journal of Hydroinformatics*, *9*(1), 25–37.

REDHISP Group. (2004). *Gisred v1.0.* `http://www.redhisp.upv.es/software/gisred/GISRed_eng.htm`. (Accessed: 2012)

rehm. (2012). *Waterpac.* `http://www.rehm.de/produkte/waterpac/default.aspx`. (Accessed: 2012)

Romano, M., & Kapelan, Z. (2014). Adaptive water demand forecasting for near real-time management of smart water distribution systems. *Environmental Modelling & Software*, *60*, 265–276.

Rossman, L. A. (2000). *EPANET 2: users manual*. US Environmental Protection Agency. Office of Research and Development. National Risk Management Research Laboratory.

Rubinstein, R. (1999). The cross-entropy method for combinatorial and continuous optimization. *Methodology and computing in applied probability*, *1*(2), 127–190.

Salomons, E., Goryashko, A., Shamir, U., Rao, Z., & Alvisi, S. (2007). Optimizing the operation of the haifa-a water distribution network. *Journal of Hydroinformatics*, *9*(1), 51–64.

Santos, C. C., & Pereira Filho, A. J. (2014). Water demand forecasting model for the metropolitan area of são paulo, brazil. *Water Resources Management*, *28*(13), 4401–4414.

Savic, D. A., & Walters, G. A. (1997). Genetic algorithms for least-cost design of water distribution networks. *Journal of water resources planning and management*, *123*(2), 67–77.

Savic, D. A., Walters, G. A., & Schwab, M. (1997). Multiobjective genetic algorithms for pump scheduling in water supply. In *Evolutionary computing* (pp. 227–235). Springer.

Schmid, R. (2002). *Review of modelling software for piped distribution networks* (2nd ed.). `http://www.skat.ch/publications/prarticle.2005-09-29.5069774463/prarticle.2006-11-02.5180575226/skatpublication.2006-11-02.2409314691`. SKAT. (Accessed: 2012)

Schneider Electric. (2012). *Citectscada - operating and monitoring software.* `http://www2.schneider-electric.com/sites/corporate/en/products-services/automation-control/products-offer/`. (Accessed: 2012)

Shihu, S., Dong, Z., Suiqing, L., Ming, Z., Yixing, Y., & Hongbin, Z. (2010). Power saving in water supply system with pump operation optimization. In *2010 Asia-Pacific Power and Energy Engineering Conference (APPEEC)* (pp. 1–4).

Simpson, A. R., Dandy, G. C., & Murphy, L. J. (1994). Genetic algorithms compared to other techniques for pipe optimization. *Journal of water resources planning and management*, *120*(4), 423–443.

SmartGridNews. (2012). *The smart water market: a slow emergence.* `http://www.smartgridnews.com/artman/publish/Technologies_Smart_Water/The-smart`

-water-market-A-slow-emergence-4905.html#.UKzW04dg-pA. (Accessed: 2012)

STANET. (2010). *Stanet - network analysis.* http://www.stafu.de/en. (Accessed: 2012)

Su, Y.-C., Mays, L. W., Duan, N., & Lansey, K. E. (1987). Reliability-based optimization model for water distribution systems. *Journal of Hydraulic Engineering*, *113*(12), 1539–1556.

Swamee, P. K., & Sharma, A. K. (2008). *Design of water supply pipe networks*. John Wiley & Sons.

Tabesh, M., & Dini, M. (2009). Fuzzy and neuro-fuzzy models for short-term water demand forecasting in Tehran. *Iranian Journal of Science & Technology, Transaction B, Engineering*, *33*(B1), 61–77.

Tahoe Design Software. (2010). *Hydroflo 2.1.* http://www.tahoesoft.com/html/hydroflo.htm. (Accessed: 2012)

Teegavarapu, R. S., & Simonovic, S. P. (2002). Optimal operation of reservoir systems using simulated annealing. *Water Resources Management*, *16*(5), 401–428.

Tiwari, M. K., & Adamowski, J. (2013). Urban water demand forecasting and uncertainty assessment using ensemble wavelet-bootstrap-neural network models. *Water Resources Research*, *49*(10), 6486–6507.

Tiwari, M. K., & Adamowski, J. F. (2014). Medium-term urban water demand forecasting with limited data using an ensemble wavelet–bootstrap machine-learning approach. *Journal of Water Resources Planning and Management*.

Tsutiya, M. T. (1997). Redução do custo de energia elétrica em estações elevatórias de água e esgoto. In *Trabalhos técnicos* (p. 15). ABES.

Tung, Y.-K. (1985). Evaluation of water distribution network reliability. In *Hydraulics and hydrology in the small computer age* (pp. 359–364).

Tynemarch Systems Engineering Ltd. (2012). *Miser - ensuring optimal water management.* http://www.tynemarch.co.uk/products/miser/miser.shtml. (Accessed: 2012)

Umweltbundesamt. (2010). *Drinking water - a scarce resource.* http://www.umweltbundesamt.de/uba-info-e/wah20-e/1-2.htm. (Accessed: 2012)

Van Zyl, J. E., Savic, D. A., & Walters, G. A. (2004). Operational optimization of water distribution systems using a hybrid genetic algorithm. *Journal of water resources planning and management*, *130*(2), 160–170.

Vasan, A., & Simonovic, S. P. (2010). Optimization of water distribution network design using differential evolution. *Journal of Water Resources Planning and Management*.

Vieira, F., & Ramos, H. (2008). Hybrid solution and pump-storage optimization in water supply system efficiency: A case study. *Energy policy*, *36*(11), 4142–4148.

Vieira, F., & Ramos, H. (2009). Optimization of operational planning for wind/hydro hybrid water supply systems. *Renewable Energy*, *34*(3), 928–936.

Viessman, W., Hammer, M. J., Perez, E. M., & Chadik, P. A. (2009). *Water supply and pollution control*. Pearson Prentice Hall New Jersey, NJ.

Vongmanee, V. (2005). The photovoltaic water pumping system using optimum slip control to maximum power and efficiency. In *2005 ieee russia power tech* (pp. 1–4).

von Lücken, C., Barán, B., & Sotelo, A. (2004). Pump scheduling optimization using asynchronous parallel evolutionary algorithms. *CLEI Electronic Journal*, *7*(2).

Walski, T. M., Brill Jr, E. D., Gessler, J., Goulter, I. C., Jeppson, R. M., Lansey, K., ... Morgan, D. R. (1987). Battle of the network models: Epilogue. *Journal of Water Resources Planning and*

Management, *113*(2), 191–203.

Walski, T. M., Chase, D. V., & Savic, D. (2001). *Water distribution modeling*. Haestad Press.

Walski, T. M., Chase, D. V., Savic, D. A., Grayman, W. M., Beckwith, S., & Koelle, E. (2003). *Advanced water distribution modeling and management*. Haestad press.

Walters, G. A., Halhal, D., Savic, D., & Ouazar, D. (1999). Improved design of the Anytown distribution network using structured messy genetic algorithms. *Urban Water*, *1*(1), 23–38.

Wang, J., & Liu, S. (2011). Quarter-hourly operation of hydropower reservoirs with pumped storage plants. *Journal of Water Resources Planning and Management*, *138*(1), 13–23.

Wang, J.-Y., Chang, T.-P., & Chen, J.-S. (2009). An enhanced genetic algorithm for bi-objective pump scheduling in water supply. *Expert Systems with Applications*, *36*(7), 10249–10258.

Wang, Y., Ocampo-Martinez, C., Puig, V., & Quevedo, J. (2014). Gaussian-process-based demand forecasting for predictive control of drinking water networks. In *Proceedings of the 9th International Conference on Critical Information Infrastructures Security, Limassol (Cyprus)*.

Wu, Z. Y., & Simpson, A. R. (2001). Competent genetic-evolutionary optimization of water distribution systems. *Journal of Computing in Civil Engineering*, *15*(2), 89–101.

Wu, Z. Y., & Simpson, A. R. (2002). A self-adaptive boundary search genetic algorithm and its application to water distribution systems. *Journal of Hydraulic Research*, *40*(2), 191–203.

Zecchin, A. C., Maier, H. R., Simpson, A. R., Leonard, M., & Nixon, J. B. (2007). Ant colony optimization applied to water distribution system design: comparative study of five algorithms. *Journal of Water Resources Planning and Management*, *133*(1), 87–92.

Zonum Solutions. (2008a). *epa2GIS*. `http://www.zonums.com/epa2gis.html`. (Accessed: 2012)

Zonum Solutions. (2008b). *EPANET Z v0.5*. `http://www.zonums.com/epanetz.html`. (Accessed: 2012)

Zonum Solutions. (2008c). *EpaSens: Epanet Sensitivity Analysis*. `http://www.zonums.com/epasens.html`. (Accessed: 2012)

Part III

Methodology and mathematical modelling

7. Hydraulic modelling

The general concepts concerning the modelling of water supply networks are presented. The methodology used by EPANET, the modelling and simulation software used in this work, is explained and a new approach for modelling variable-speed pumps is proposed and compared.

The worldwide concerns with economic and environmental sustainability increase the importance of a detailed understanding on the operation of certain systems in order to control them efficiently (on both energetic and economic points of view). This is the case of the water supply systems that deal with large amounts of energy usually responsible for a large portion of the total costs associated to their operation.

Since the water networks are very complex systems, understanding some aspects in their operation and, at the same time, dealing with a large number of elements such as multiple pumps, valves and tanks is a difficult task. Consequently, workers and researchers in the water field often rely on computer programmes for supporting the management, operation and analysis of water systems.

Simulation models allow the computational representation/reproduction of real systems behaviour. Therefore, the use of such models for the operational efficiency analysis of water supply networks is, in fact, the most common practice among researchers and students of hydraulic engineering.

7.1 Modelling water networks

Water supply and distribution networks are characterised by interconnected hydraulic elements including pipes, junctions, tanks, reservoirs, pumps and valves. The connections between such elements are defined by the principles of mass and energy conservation (Walski, Chase, & Savic, 2001).

According to the computational models representation, networks are comprised of nodes, which represent specific locations, and links, which define the relationship between nodes (Walski et al., 2001). Typically, junctions, tanks and reservoirs are represented as nodes and pipes are represented as links. In turn, pumps and valves can be represented by nodes or links, depending on the defined methodology.

Reservoirs represent infinite sources of water, *i.e.* elements with a large capacity that keeps the hydraulic grade constant. These elements intend to represent lakes, groundwater wells or even treatment plants (Walski et al., 2001). The information required to model this type of element is the

water surface elevation (hydraulic grade line or hydraulic head) and the water quality (when water quality analyses are required).

Tanks, or storage reservoirs, are finite sources of water. While in steady-state simulations the tank is hydraulically identical to a reservoir for a given hydraulic grade line, in extended period simulations, the water level in the tank is allowed to vary over time, influencing the hydraulic grade line. Thus, the necessary input properties to model a tank include: the elevation*, the tank dimensions, the initial, minimum and maximum water levels and, when necessary, the initial water quality.

Junctions typically represents the location where two or more pipes met. However, it may also represent the end of a single pipe (Walski et al., 2001). Only the elevation needs to be specified by the modeller. This kind of element can also be used to model a point with an associated water demand or a specific water injection in the network (negative demand). In this case, a demand pattern should also be specified.

Pipes are the elements that transport the water from one node to another. The main necessary characteristics to model a pipe are: length, diameter and material.

Valves are elements that can be open and closed in order to limit the flow in a certain part of the network, controlling the movement of the water. Valves are generally classified in five main categories (Walski et al., 2001):

- Isolation valves (gate, butterfly, globe and plug valves), are used to block the flow of water for maintenance or emergencies. In most cases, it is not necessary to include such valves in hydraulic models. Instead, their influence in the network can be included in the pipes properties (through minor losses).
- Directional valves (or check valves), used to ensure the water flow through only one direction (*i.e.* to avoid back-flow). Can also be modelled by editing the pipes properties.
- Altitude valves, normally installed in the entrance of the tanks (inflow) in order to avoid the entrance of more flow when the tank is full (overflow). In modelling computer programmes, such behaviour is usually incorporated in the tanks.
- Air release or vacuum breaking valves, used to release trapped air or discharge air (to respond to negative pressures) in the system. Such elements are usually not included in the networks. Their main purpose is for advanced studies such as transient analysis.
- Control valves (or regulating valves), used to regulate flow, throttle (minor loss) or pressure in the system. Hydraulic models typically support this type of valves, including:
  - Flow Control Valves (FCV) that maintain the flow rate under a user-specified limit;
  - Throttle Control Valves (TCV) that throttle to adjust the minor loss coefficient based on some attribute of the system (a critical nodal pressure, a tank level, etc.).
  - Pressure Reducing Valves (PRV) that automatically throttle the flow to prevent the hydraulic grade from exceeding a certain value and avoid excessive pressures in specific zones of the network. For this reason, these are typically installed between two pressure zones.
  - Pressure Sustaining Valves (PSV) that also automatically throttle the flow to prevent the hydraulic grade from dropping below a certain value. Such element is then characterised by the upstream pressure it tries to maintain.

*Should not be confused with the hydraulic head, which, in this case, corresponds to the water surface elevation and is obtained as a model output.

Pumps are elements that provide hydraulic energy to the system (the mechanical energy of the rotating impeller is imparted to the water), increasing the hydraulic grade. This kind of element is typically used to overcome piping headlosses and physical elevation differences. In water distribution systems, the most common used pumps are centrifugal (Walski et al., 2001). To model a pump, only the characteristic head curve (*i.e.* head added by the pump, H, versus discharge, Q) is necessary. However, to compute the energy consumption, the associated efficiency curve should also be included in the model (also as a function of the discharge). Pumps operating at variable-speed produce different head and discharge characteristics that should also be taken into account when modelling the systems.

After defining the basic elements of a network and its topology, the integral formulation model can be refined depending on its purpose. The two main types of hydraulic simulation typically performed are: (i) the steady-state simulation and (ii) the extended period simulation. While the first one assumes no changes in the system with time, providing only the state of the modelled system, the second one computes a kind of dynamic behaviour of the system over a period of time in order to predict changes in the system. Thus, in this last case, some additional information should be provided to the model, including (Walski et al., 2001): (i) the simulation duration (typically, a multiple of 24 hours is used, however, it is possible to simulate an entire week or more), (ii) the hydraulic time step, which should not be too large in order to avoid abrupt hydraulic changes (1-hour time steps are normally acceptable, however, the simulation accuracy can be improved by reducing the step), and (iii) specific changes that do not occur in time increments.

7.1.1 Principles of mass and energy conservation

In fluid dynamics, pipe network analysis is the integral formulation analysis of the fluid flow through an hydraulic network in order to determine the flow rates, heads and pressures in specific locations of the network. In order to perform such analysis, the principles of mass and energy conservation are applied to each element of the network.

The principle of Conservation of Mass (or flow continuity) states that the fluid mass that enters into a pipe (inflow) is equal to the mass that leaves the pipe (outflow). In networks modelling, the flow continuity is considered for each node, instead of pipes (Walski et al., 2001). Thus, according to this principle, the difference between a node i inflow from node j, Q_{ij} (m^3/s), and the node i outflow (or demand at node i), Q_i^{out} (m^3/s), should be zero, which gives:

$$\sum_j Q_{ij} - Q_i^{\text{out}} = 0, \qquad \text{for} \qquad i = 1, ..., n_{\text{nodes}}, \tag{7.1}$$

where n_{nodes} represents the number of nodes.

Specifically for tanks, in extended period simulations, the changes in storage should also be taken into account and Equation 7.1 should be re-written as (Walski et al., 2001):

$$\sum_j Q_{ij} - Q_i^{\text{out}} - \frac{\mathrm{d}S}{\mathrm{d}t} = 0, \qquad \text{for} \qquad i = 1, ..., n_{\text{nodes}}, \tag{7.2}$$

where $\frac{\mathrm{d}S}{\mathrm{d}t}$, in m^3/s units, represents the change in storage in the time interval t.

The principle of Conservation of Energy states that the difference in energy between two points should be the same. In hydraulic analysis, the equation that describes this principle is written in terms

of head. Thus, in a link (or links) between nodes i and j, the equation that describes the energy conservation (also known as the Bernoulli equation) can be given by (Walski et al., 2001):

$$Z_i + \frac{p_i}{\gamma} + \frac{v_i^2}{2g} + \sum^{n_{\text{pumps}}} h_{\text{p}} = Z_j + \frac{p_j}{\gamma} + \frac{v_j^2}{2g} + \sum^{n_{\text{pipes}}} h_{\text{L}} + \sum^{n_{\text{pipes}}} h_{\text{m}}, \tag{7.3}$$

where Z is the elevation, p is the pressure, γ is the fluid specific weight, v is the fluid velocity, $g = 9.81$ m/s^2 is the gravitational acceleration, h_{p} is the head added at pumps, h_{L} is the headloss in pipes and h_{m} is the headloss due to minor losses. This shows that the energy difference between the two nodes is equal to the energy gains from pumps and energy losses in pipes (and fittings) that occur in the path between them (Walski et al., 2001), *i.e.*

$$H_j - H_i = \sum^{n_{\text{pumps}}} h_{\text{p}} - \sum^{n_{\text{pipes}}} h_{\text{L}} - \sum^{n_{\text{pipes}}} h_{\text{m}}, \tag{7.4}$$

where H_j and H_i are, respectively, the total head (or energy) in nodes j and i.

In hydraulic models, according to the previous described equations, typically one continuity equation is developed for each node and one energy equation is developed for each link in the network (Walski et al., 2001). The method used to solve the set of equations can vary between hydraulic simulators.

7.1.2 EPANET methodology

EPANET is the most widely used free software for simulation of pressurised water networks. Besides being a Windows® programme freely distributed in the public domain, EPANET comes with a Programmer's Toolkit, a dynamic link library (DLL) of functions that allow developers to customise EPANET to their own needs (*US EPA: EPANET*, 2015), which was used to incorporate the code developed in this work using the programming language C/C++.

In EPANET, pumps and valves are represented by links. In the case of pumps, both constant- and variable-speed pumps can be modelled. To each pump, a characteristic operating curve (head versus flow) should be assigned. It is also possible to assign an efficiency curve and a pattern of the speed settings (for variable-speed pumps). By default, a relative speed setting of 1 is associated to the pump characteristic curve supplied by the user. In the definition of a speed pattern, if the user intends, for instance, to double the pump speed, the setting should be set to 2. On the other side, if the pump is intended to run at half the speed, the setting should be set to 0.5. A schedule of energy prices can also be assigned for the energy cost computation (Rossman, 2000).

With respect to the valves, an EPANET model supports the four main types of control valves (PRV, PSV, FCV and TCV) and also a General Purpose Valve (GPV), used to model special flow-headloss relationships required by the users. This last type of valve is typically used to model turbines (Rossman, 2000). Check valves are modelled through pipes by changing their status. Isolation valves can be modelled by changing the minor loss coefficient associated to a pipe according to the adequate value for each type of valve.

Minor headlosses (or local losses), associated to pipes, are used to represent the added turbulence in the systems related to bends and fittings (such as valves, elbows, tees, reducers). The equation to

compute such losses is given by (Rossman, 2000):

$$h_{\mathrm{m}} = K_{\mathrm{m}} \left(\frac{v^2}{2g} \right), \tag{7.5}$$

where K_{m} represents a dimensionless minor loss coefficient, v is the fluid velocity and g is the gravitational acceleration. The minor headlosses, h_{m}, are given in length units.

The headlosses due to the fluid friction with the pipe walls can be computed by selecting one of three available formulas: (i) the Hazzen-Williams, (ii) the Darcy-Weisbach or (iii) the Chezy-Manning formula. The Chezy-Manning (C-M) formula is not commonly used for water distribution modelling but typically for open channel flow. The Hazzen-Williams (H-W) formula, an empirically-based formula, is the most used in U.S. and can only be used when the fluid is water and typically under turbulent flow. On the other hand, the Darcy-Weisback (D-W) equation, more used in Europe, is a physically-based formula developed using dimensional analysis and can be applied to all liquids over all flow regimes (Rossman, 2000; Walski et al., 2001).

In EPANET, the friction headlosses between the start and the end node of a pipe, h_{L}, are computed using the general formula (Rossman, 2000):

$$h_{\mathrm{L}} = a Q_{ij}^{b}, \tag{7.6}$$

where Q_{ij} is the flow rate between the start and end nodes, i and j, and a and b are, respectively, the resistance coefficient and the flow exponent. These last two variables are computed according to the headloss formula considered. Using the H-W formula, Equation 7.6 becomes:

$$h_{\mathrm{L(H\text{-}W)}} = \frac{C_f L}{C^{1.852} D^{4.87}} Q_{ij}^{1.852}, \tag{7.7}$$

where L is the length of the pipe, C is the dimensionless Hazen-Williams roughness coefficient[†] (or C-factor), D is the pipe diameter and C_f is a unit conversion factor (in S.I. units $C_f = 10.7$). In turn, if the D-W formula is considered, Equation 7.6 takes the form:

$$h_{\mathrm{L(D\text{-}W)}} = \frac{8 f(Re, \frac{\varepsilon}{D})}{\pi^2 g} \frac{L}{D^5} Q_{ij}^2, \tag{7.8}$$

where f is the dimensionless Darcy-Weisback friction factor (also called Moody diagram friction factor) that is function of the Reynolds number, $Re = \frac{vD}{\nu}$ (where ν is the kinematic viscosity), the Darcy-Weisback roughness coefficient, ε, and the pipe diameter, D. For laminar flow ($Re \leq 2000$), EPANET determines f by the Hagen-Poiseuille formula, $f = \frac{64}{Re}$ (Rossman, 2000). For turbulent flow ($Re \geq 4000$), the following approximation to the Colebrook-White equation is used (Rossman, 2000):

$$f = \frac{0.25}{\left[\ln\left(\frac{\varepsilon}{3.7D} + \frac{5.74}{Re^{0.9}} \right) \right]^2}. \tag{7.9}$$

Finally, for flow in the transition regime ($2000 < Re < 4000$), EPANET performs a cubic interpolation from Moody Diagram[‡] (for more detail on this, see Rossman, 2000).

[†] See C-factors for various pipe material in Table C.1 of Appendix C.

[‡] Moody Diagram available in Figure C.1 of Appendix C.

As previously stated, a GPV is typically used to model turbines. Instead of following a standard hydraulic formula, a GPV allows the user to provide a curve representing the intended headloss relationship with the flow through the link. Another possible way to model turbines is by means of emitters. This method can be used for water excess turbines, such as Pelton turbines with free outflow (Sitzenfrei, Berger, & Rauch, 2015). An emitter is defined as a junction property in EPANET and not as a separate network component. The user should provide the value of the emitter coefficient that represents the flow that occurs at a pressure drop of 1 m (or psi). The flow rate through the emitter, Q, will then vary as a function of the pressure head P_{man} (or manometric pressure) available at the junction node according to (Rossman, 2000):

$$Q = C_{\text{e}} P_{\text{man}}^{s}, \tag{7.10}$$

where C_{e} represents the emitter coefficient and s is the pressure exponent. Typically $s = 1/2$.

When an emitter coefficient is associated to a junction node, the emitter is modelled as a fictitious pipe between the junction and a fictitious reservoir. The pipe's headloss coefficients (see Equation 7.11) are given by $a = (1/C_{\text{e}})^{1/s}$, $b = 1/s$ and $K_{\text{m}} = 0$. The head at the fictitious reservoir is the elevation of the junction node and the computed flow through the fictitious pipe becomes the flow associated with the emitter (Rossman, 2000).

EPANET solves the flow continuity and energy equations that characterise the hydraulic state of all nodes and links in the network at each defined time period in which the simulation time is divided. For pipes, the headlosses between nodes i and j for each time-step are determined by the flow-headloss relation (Rossman, 2000):

$$H_i - H_j = h_{\text{L}} + h_{\text{m}} = a Q_{ij}^{b} + K_{\text{m}} Q_{ij}^{2}. \tag{7.11}$$

For pumps, the equation that describes the relation between the pump headloss (negative of the head gain) and flow is mathematically defined as Rossman (2000):

$$h_{\text{p}}^{\text{loss}} = -M^2 h_0 + M^2 r \left(\frac{Q_{ij}}{M} \right)^{n}, \tag{7.12}$$

where M is a relative speed setting, r and n are pump curve coefficients, and h_0 is the pump shut-off (or cut-off) head.

The set of equations that result from the mass and energy conservation in all elements of the network (*i.e.* Equations 7.2, 7.11 and 7.12) are dependent on each other, leading to a system of equations where the unknowns are the nodal heads, H, the pumps head gains, h_{p}, the pumps flow rates, Q, and the pipes headlosses, h_{L}. To solve the systems of equations, EPANET follows a node-loop approach, also called gradient method (Rossman, 2000).

The gradient method starts with an initial estimation of pipe flows that may or not satisfy the flow continuity. Then, iteratively, new nodal heads are computed by solving the matrix equation (Rossman, 2000):

$$\mathbf{A}\mathbf{H} = \mathbf{F}, \tag{7.13}$$

where $\mathbf{H}$ is a matrix of size $n_{\text{nodes}} \times 1$ of the unknown nodal heads, $\mathbf{A}$ ($n_{\text{nodes}} \times n_{\text{nodes}}$) is a Jacobian

matrix and $\mathbf{F}$ ($n_{\text{nodes}} \times 1$) is a matrix of the right hand side terms. The diagonal elements of matrix $\mathbf{A}$ are:

$$A_{ii} = \sum_j B_{ij}, \tag{7.14}$$

and the non-zero elements outside the diagonal are:

$$A_{ij} = -B_{ij}, \tag{7.15}$$

where B_{ij} is the inverse of the derivative of the headloss in links between node i and j (described by Equations 7.11 and 7.12). For pipes,

$$B_{ij} = \frac{1}{ab|Q_{ij}|^{b-1} + 2K_{\text{m}}|Q_{ij}|}, \tag{7.16}$$

and for pumps,

$$B_{ij} = \frac{1}{nM^2 r\,(Q_{ij}/M)^{n-1}}. \tag{7.17}$$

The elements of matrix $\mathbf{F}$ are given by an equation where a correction factor is added to the flow balance at a node:

$$F_i = \left(\sum_j Q_{ij} - Q_i^{\text{out}}\right) + \sum_j q_{ij} + \sum_f B_{if} H_f, \tag{7.18}$$

where the last term applies to any link connecting node i to a node f with a fixed-grade H_f. For pipes, the flow correction factor q_{ij} is given by:

$$q_{ij} = B_{ij}\left(a|Q_{ij}|^b + K_{\text{m}}|Q_{ij}|^2\right)\text{sng}\,(Q_{ij}), \tag{7.19}$$

and for pumps:

$$q_{ij} = -B_{ij}\left[M^2 h_0 - M^2 r\,(Q_{ij}/M)^n\right], \tag{7.20}$$

where $\text{sng}\,(Q_{ij})$ is 1 if Q_{ij} is positive or -1 if negative. For pumps, Q_{ij} is always positive (Rossman, 2000).

After computing the new nodal heads by solving the matrix equation 7.13, the flows are updated by (Rossman, 2000):

$$Q_{ij} = Q_{ij} - [q_{ij} - B_{ij}\,(H_i - H_j)], \tag{7.21}$$

which always results in flow continuity around each node after the first iteration.

Equations 7.13 and 7.21 are repetitively solved until the sum of absolute flow changes relative to the total flow in links reaches a value smaller than a tolerance value (*e.g.* 0.001). For more detail on the EPANET iterative method see Rossman (2000).

Besides the network analysis, EPANET is also able to compute the energy consumption associated to each pump operation during a certain simulation period. The electric power consumed by each pump is computed by:

$$P = \gamma\frac{QH}{\eta}, \tag{7.22}$$

where Q and H are, respectively, the flow discharge and head gain, $\gamma = \rho g$ is the specific weight of the water (with ρ representing the water density) and η is the overall (wire-to-water) pump efficiency.

In the methodology of EPANET, if instead of considering a constant efficiency, an efficiency curve is assigned to the pump, then, the value of the pump efficiency at each time-step of the simulation is determined by an interpolation of the values of such curve for the corresponding pump flow. However, it is important to note that EPANET always considers the same efficiency curve, even for distinct pump speeds which, in some cases, does not correspond to good approximations of real pumps behaviour. This particular topic is discussed in section 7.2.

It must be noted that the overall efficiency, η, presented in Equation 7.22, is given by:

$$\eta = \eta_{\mathrm{m}}.\eta_{\mathrm{VFD}}.\eta_{\mathrm{p}}, \tag{7.23}$$

where η_{p} corresponds to the pump efficiency, η_{m} is the efficiency of the motor coupled to the pump and η_{VFD} corresponds to the efficiency of the variable frequency drive, which can be considered equal to 1 when no drive is used.

Since the pump energy consumption (E) can be determined for each time period by the multiplication between the power (P) and the pump operating time, t_{op}, *i.e.* $E = P \times t_{\mathrm{op}}$, then, the total energy consumption associated to a pipe network operation can be determined by:

$$E_{\mathrm{total}} = \gamma \sum_{p=1}^{n_{\mathrm{pumps}}} \sum_{s=1}^{n_{\mathrm{steps}}} \left(\frac{H_{p,s} Q_{p,s}}{\eta_{p,s}} \times t_{\mathrm{op},p,s} \right), \tag{7.24}$$

where n_{steps} is the number of time-steps in which the simulation is divided and n_{pumps} is the number of pumps in the network.

7.2 Modelling variable-speed pumps

Pumps are essential components on energy efficiency studies of WSS. A pump is a device that transfers the mechanical energy to the fluid as hydraulic head. This head, called pump head, is a function of the flow that passes through the pump. Thus, the pumps are used when the WSS needs energy to overcome elevation differences. Centrifugal pumps are the mostly used in this kind of system (Walski et al., 2001).

The relationship between a pump head gain, H, and discharge, Q, is represented by the pump head characteristic curve defined by the general equation:

$$H = h_0 - rQ^n, \tag{7.25}$$

where h_0 represents the pump shut-off head and r and n represent the curve coefficients. This is a non-linear curve that shows a decreasing head with the flow rate through the pump.

Pumps can be of constant or variable speed and should operate inside the limits imposed by their characteristic head curves.

Modelling the behaviour of pumps operating at variable-speed resulting from the use of Variable Frequency Drives (VFDs) implies the adaptation of the pump characteristic curve and the power and hence the efficiency curves for each distinct speed. For the prediction of such curves, the affinity laws are commonly used (Rossman (2000), Quintela (1981), Walski, Zimmerman, Dudinyak, and Dileepkumar (2003), etc.). Such laws reflect the fact that dimensionless characteristics, such as the

dimensionless flow (C_Q), the dimensionless head (C_H) and the dimensionless power (C_P), are constant for similar pumps (Simpson & Marchi, 2013).

Derived from the dynamic similitude (or dimensionless representation of test results), considering single-phase liquid flow and negligible viscous forces, the dimensionless pump characteristics relate flow, head and power to the speed and the impeller diameter of the pump according to the following expressions (Martin, 2000):

$$\text{(a)} \quad C_Q = \frac{Q}{\omega D_p^3}, \qquad \text{(b)} \quad C_H = \frac{gH}{\omega^2 D_p^2}, \qquad \text{and} \qquad \text{(c)} \quad C_P = \frac{P}{\rho \omega^3 D_p^5}, \tag{7.26}$$

where ω is the angular velocity of pump, D_p is the impeller diameter and ρ is the liquid density (water density, in this case). The efficiency, η, is indirectly described by the equation of the pump power consumption, previously presented (Equation 7.22).

For variable-speed pumps, considering that the impeller diameter is kept constant and only the pump speed is modified, then the affinity laws can be derived from equations 7.26a to 7.26c. These laws show that pump flow, head and power are, respectively, linear, quadratic and cubic functions of the pump speed:

$$\text{(a)} \quad \frac{Q_1}{Q_2} = \frac{N_1}{N_2}, \qquad \text{(b)} \quad \frac{H_1}{H_2} = \left(\frac{N_1}{N_2}\right)^2, \qquad \text{and} \qquad \text{(c)} \quad \frac{P_1}{P_2} = \left(\frac{N_1}{N_2}\right)^3, \tag{7.27}$$

where N_1 and N_2 correspond to two different pump speeds ($N = (\omega/2\pi) \times 60$ is the rotational speed, in rpm). The described laws consider that the value of the pump efficiency at the best efficiency point (BEP) is kept constant with the speed variation. The efficiency curve is only moved to the left when the pump speed is reduced or moved to the right when increased (Marchi & Simpson, 2013).

Thus, for variable-speed pumps, Equation 7.25 must be modified according to the affinity laws for flow and head, provided in equations 7.27a and 7.27b. Changing the pump speed from N_1 to N_2, the modified characteristic curve can be obtained by replacing H_1 and Q_1 (head and flow at speed N_1) with the expressions given for the affinity laws, which leads to:

$$H_2 = h_0 \left(\frac{N_2}{N_1}\right)^2 - r\left(\frac{N_2}{N_1}\right)^2 \left[\frac{Q_2}{\left(\frac{N_2}{N_1}\right)}\right]^n, \tag{7.28}$$

which is equivalent to the equation used by EPANET for the headloss[§] and flow relationship for pumps (equation 7.12). Figure 7.1 shows an example of a pump characteristic curve change with speed variation.

Understanding the effect of the speed variation in the pump design curves (head, power and efficiency) is important for a real perception of the savings resulted from the installation of variable-speed drives or from the replacement of fixed-speed pumps by variable-speed pumps.

According to Martin (2000), the affinity laws for flow discharge and head are accurate since they are based on actual tests for all types of centrifugal pumps. On the other hand, as efficiency increases with the size of the pump, the affinity law for power is not so accurate. In fact, the main limitations of the affinity laws are related to factors that do not scale with velocity and whose magnitude depends

[§] In fact, the equation is symmetric to equation 7.12, since, in this case, it represents the pump head gain and not the headloss.

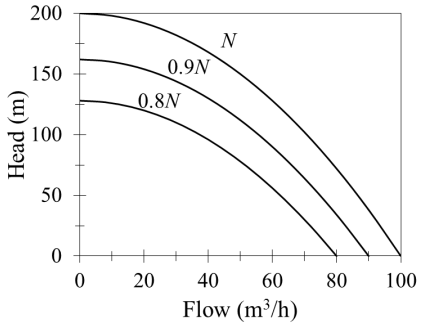


Figure 7.1: Example of a pump characteristic curve change with speed reduction in 90 % and 80 % of the nominal speed N.

on the machine size (Simpson & Marchi, 2013).

Morton (1975) and Sárbu and Borza (1998) provided charts showing an approximation on how speed variation affects the design flow, head, power and/or efficiency of centrifugal pumps. Similarly to the presented by these authors, Figure 7.2 provides graphical relationships between the ratio of each pump design characteristic with the ratio of speed reduction from the nominal speed. The representations for both pump flow and head are exactly in agreement with the presented in the works of Morton (1975) and Sárbu and Borza (1998), showing the flow varying directly with speed and the head with the square of speed, which also meets the affinity laws for flow and head (equations 7.27a and 7.27b).

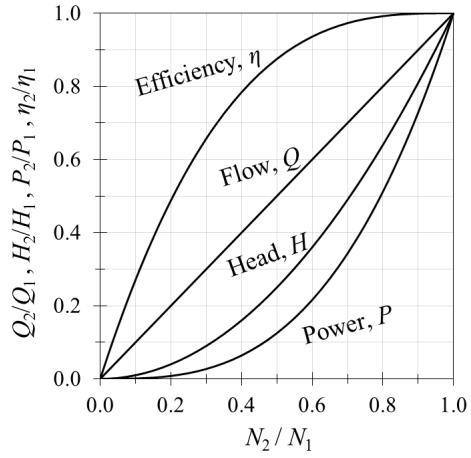


Figure 7.2: Approximated representation of the variation of the design head, flow, power and efficiency of a centrifugal pump when varying the speed from its nominal speed N_1 (adapted from Morton, 1975 and Sárbu & Borza, 1998).

Concerning the pump power, the curve represented in Figure 7.2 reflects the cubic variation of the pump power with the percentage of speed decrease, such as the represented by the affinity law for power in equation 7.27c and also shown by Sárbu and Borza (1998) in their graphical representation of the pump power affected by speed reduction.

The pump efficiency curve, in turn, requires a more detailed analysis. Reproducing the pump efficiency curves graphically presented in the works of Morton (1975) and Sárbu and Borza (1998), such as the represented in Figure 7.3, it is possible to observe that the curves are not coincident. In

fact, while Morton (1975) only provided a graphical approach for the efficiency prediction, Sárbu and Borza (1998) also related the speed-adjusted efficiency (η_2) with the original efficiency (η_1) through the following equation:

$$\eta_2 = 1 - (1 - \eta_1)\left(\frac{N_1}{N_2}\right)^{0.1}. \tag{7.29}$$

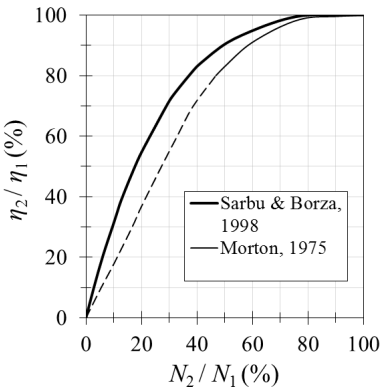


Figure 7.3: Reproduction of the curves provided in the works of Morton (1975) and Sárbu and Borza (1998) for the representation of the pumps efficiency variation with speed.

Morton (1975) stated that this prediction of pump efficiency (referred as an estimate for average conditions) should only be considered accurate for pump speeds between the full-speed and 50 % of the full-speed. Below those speed values (such as the represented by the efficiency dashed line in Figure 7.3), the efficiency reduction tends to be greater than the predicted for small pumps and lower for large pumps.

As already mentioned by Simpson and Marchi (2013), equation 7.29 can be considered as an approximation to the equation presented by Gulich (2003) of the efficiency estimation for hydraulically smooth surfaces, which considers the effect of the Reynolds number (*Re*) and assumes that only a fraction V of the energy losses are *Re* dependent:

$$\frac{1-\eta_2}{1-\eta_1} = V + (1-V)\left(\frac{Re_1}{Re_2}\right)^{m_{Re}}, \tag{7.30}$$

where Re_1 and Re_2 correspond to the Reynolds number at the speeds N_1 and N_2, respectively. The exponent m, dependent on the Reynolds number and roughness, can vary between 0 for the fully rough (turbulent) flow region and 1 for laminar flow. Gulich (2003) pointed out that other authors found $V = 0$ to 0.57 and $m_{Re} = 0.1$ to 0.5 in experimental data. Equation 7.30 can become equation 7.29 assuming $V = 0$ (energy losses not dependent on *Re*), $m = 0.1$ and replacing Re_1/Re_2 by N_1/N_2, since the fluid velocity is proportional to the pump speed (Simpson & Marchi, 2013).

The advantage of the equation proposed by Sárbu and Borza (1998) (equation 7.29) for the efficiency prediction is related with the ability to predict distinct efficiency curves for distinct original efficiency points (η_1), as demonstrated in Figure 7.4.

Sárbu and Borza (1998) mentioned that, especially for large pumps, the changes in efficiency can be neglected for rotational speeds reductions until $1/3$ of the nominal speed. Observing Figure 7.7, it can be also noted that, according to the SB equation (Equation 7.29), the reduction in efficiency with

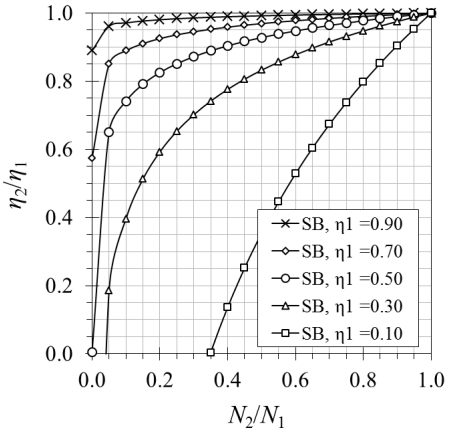


Figure 7.4: Curves predicted with the formulation proposed by Sárbu and Borza (1998) for the efficiency change with speed, considering different original efficiencies (η_1).

the speed decrease is greater for lower original efficiencies. However, as already stated by Sárbu and Borza (1998), their formulation can lead to negative values of the efficiency, which may represent a disadvantage.

In this work, a distinct formulation for the efficiency curves prediction with speed variation that overcomes the main drawback stated by Sárbu and Borza (1998) for their formulation is proposed. The method used to derive such formulation is explained in the following subsection.

7.2.1 Proposing a new formulation for the speed-adjusted efficiency curves

Despite the affinity laws assume that the BEP is maintained with the pump speed reduction, in real pumps operation, a reduction in its efficiency is verified when decreasing the rotational speed.

Trying to obtain an equation to describe a similar graphical representation of the efficiency variation with speed with the presented by both Morton (1975) and Sárbu and Borza (1998) (Figure 7.3), a theoretical formulation derived from the graphical representation of the pump power function of Figure 7.2, is proposed (Coelho & Andrade-Campos, 2016).

In Figure 7.5, the curve that represent the cubic variation of a pump power (P) with speed variation (N) is shown in quadrant I (see the grey area). Extending this power function to quadrant III, the shape of the observed curve is in accordance with the curve shape expected for the efficiency variation with speed. Thus, starting from the function that describes the power variation, $P(N)$, performing an horizontal translation to such function (step A) and finally, applying a vertical translation to the obtained function (step B), leads to a function that describes a pump efficiency variation with speed changes. Mathematically, the described translations can be represented as:

$$\eta(N) = P(N-1) + 1. \tag{7.31}$$

The variables P, N and η were used for simplification in order to explain how the proposed formula was derived from the graph. Replacing them by the proper expressions, $P = P_2/P_1$, $N = N_2/N_1$ and $\eta = \eta_2/\eta_1$, and considering the affinity law for power (Equation 7.27c), then Equation

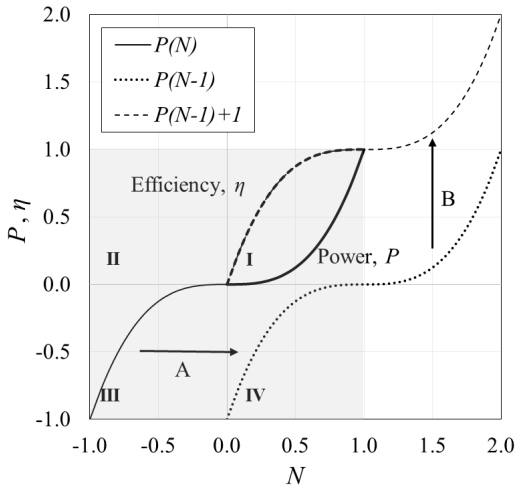


Figure 7.5: Representation of the translations of the power curve, $P(N)$, that lead to the proposed formulation for the efficiency curves prediction. $P(N-1)$ is an horizontal translation of $P(N)$ and $P(N-1)+1$ is a vertical translation of $P(N-1)$.

7.31 can be expressed as:

$$\frac{\eta_2}{\eta_1} = \left(\frac{N_2}{N_1} - 1\right)^3 + 1, \tag{7.32}$$

which reveals a new equation for speed-adjusted efficiency curves of centrifugal pumps (Coelho & Andrade-Campos, 2016).

Figure 7.6 demonstrates that the efficiency curve proposed in this work approximates the curves presented in the works of Morton (1975) and Sárbu and Borza (1998). However, besides this new formulation never reaches negative values, it can be seen in Figure 7.7 that, independently on the original efficiency considered, the change in efficiency with speed remains the same.

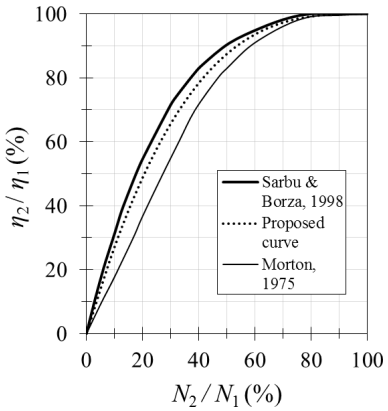


Figure 7.6: Curve representing the relationship proposed in this work for the efficiency reduction with speed decrease compared with the ones proposed by (Morton, 1975) and by (Sárbu & Borza, 1998).

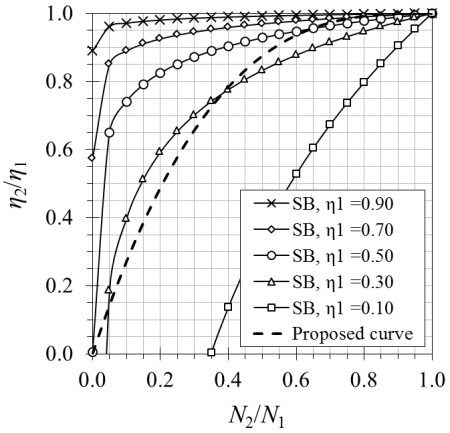


Figure 7.7: Precticition of the efficiency variation with the speed reduction, considering distinct original efficiencies, η_1, using the (Sárbu & Borza, 1998) equation (SB) and comparison with the curve proposed in this work.

8. Control optimisation

The formulation of the optimisation problem according to the proposed approach is presented. The optimisation algorithms with relevance for this work are explained, particularly the Nelder-Mead Simplex, the Genetic Algorithms, the Differential Evolution, the Particle Swarm Optimisation and the Simulated Annealing. Penalty-based constraint-handling methods are also described.

Nowadays, the major expenses in water supply systems (WSS) are related to energy consumption. The fast expansion of several water supply systems due to the population growth and the immediate consumers supply without any planned strategy have led to inefficiently operated systems. In these systems, pump stations usually represent the main operational costs (Van Zyl, Savic, & Walters, 2004; Vieira & Ramos, 2009) revealing an important opportunity for the efficiency improvement of the water supply systems. In fact, the demand by industry to control pump systems efficiently has been increasing.

Generally, in most WSS, the pump stations operations are only based on the pumps on/off, without taking advantage of variable-speed equipment. The pumps are switched on when the tanks, responsible for supplying the populations, reach their minimum levels. These pumps are only switched off when the tanks reach their maximum levels. In the cases where variable-speed equipment is used, planned control strategies are almost always missing. The introduction of operational pump schedules adapted to the energy prices variation and to the consumption patterns of the populations can optimise pumping stations operations, minimising energy consumption and costs significantly. However, the process of finding the best pump pattern can present difficulties due to the complexity of some WSS (multiple pumps, multiple reservoirs, non-linear behaviour of the systems, etc.). At the same time, water supply and distribution systems should satisfy the consumption, attending to the demand in each place and time and with appropriate pressures (Viessman, Hammer, Perez, & Chadik, 2009). All these factors need to be taken into account during any operational control procedure, which increase the difficulty of the task.

8.1 Optimisation approach and problem formulation

In this work, a methodology for the optimal control of pumps and valves in water supply and distribution systems is proposed. The proposed methodology is based on an explicit control optimisation problem where the decision variables include (i) the pumps operating times and relative speed (if

applicable) as well as (ii) the valves operating times (opening time periods). The idea is to simultaneously optimise the control of all pumps (of variable- and fixed-speed) and valves of a network, taking into account the variation of the energy price and the network requirements. Both the variables operating time, $t_{op} \in [0, t_{step}]$ (where t_{step} is the step size), and relative speed, $M \in [M_{min}, M_{max}]$ (where $M = N_2/N_1$), are scaled (normalised) in the range $[0, 1]$, according to the equation:

$$x_{scaled} = \frac{x_{non\text{-}scaled} - x_{min}}{x_{max} - x_{min}}, \tag{8.1}$$

where x_{scaled} represents the scaled variable and x_{min} and x_{max}, the maximum and minimum values of the non-scaled variable $x_{non\text{-}scaled}$. This procedure is particularly useful when dealing with variables of different units and scales. Moreover, this normalisation procedure has the advantage of not restricting the operating times to a certain fixed step size (such as the typical 1-hour steps), being adaptable to any dimension pre-defined by the user for the simulation step size. With this approach, instead of considering, for instance, typically fixed 1-hour time-steps for the optimisation procedure, the operating time of a pump is allowed to be inferior to the pre-defined hydraulic step of the model. It is also allowed to present different operating times in each time-step of the simulation. The same approach is followed for the valves settings (open/closed). Instead of considering fixed 1-hour opening times (fully or partially open), the valves are allowed to be open (fully open, in this case) during time periods inferior to the defined hydraulic step, which, in terms of hydraulic results, leads to a similar solution since the hydraulic computation is performed for each step.

A single-objective optimisation approach is considered, where the purpose is to reduce the energy costs associated to the network operation. Since each network, depending on its configuration and on the regulation (that vary for distinct countries) presents for specific operating conditions distinct requirements, then the fulfilment of such requirements is ensured by the use of penalties.

Mathematically, a general optimisation problem can be described by the minimisation (or maximisation) of a function f (objective function) subject to the bounds of n_{var} decision variables ($\mathbf{x}^{min}$ and $\mathbf{x}^{max}$) and subject to J inequality constraints (g_j) and/or K equality constraints (h_k):

$$\begin{array}{lll} \underset{\mathbf{x}}{\text{minimise}} & f(\mathbf{x}), & \mathbf{x} = [x_1 \;\; x_2 \;\; \ldots \;\; x_{n_{var}}]^T; \\ \text{subject to:} & g_j(\mathbf{x}) \leq 0, & j = 1, \ldots, J, \\ & h_k(\mathbf{x}) = 0, & k = 1, \ldots, K; \\ & x_i^{min} \leq x_i \leq x_i^{max}, & i = 1, \ldots, n_{var}. \end{array} \tag{8.2}$$

Considering that the main goal in the present study is to minimise the costs associated to water pumping in a WSS, the objective function corresponds to the costs associated to the pumps operation. This cost is calculated through the use of the hydraulic simulator EPANET 2.0 (Rossman, 2000).

During the hydraulic analysis of each network model, EPANET solves the flow continuity and headloss equations as previously explained in Chapter 7. Considering the pumping energy computation formula used by EPANET (see equation 7.24), the objective function can be obtained by the cost associated to such energy consumption taking into account the energy tariff associated to each pump,

which leads to:

$$\begin{aligned} \mathrm{C_{total}} &= \mathrm{C_{pumping}} + \mathrm{C_{demandcharge}} \\ &= \gamma \sum_{p=1}^{n_{\mathrm{pumps}}} \sum_{s=1}^{n_{\mathrm{steps}}} \left(\frac{H_{p,s} Q_{p,s}}{\eta_{p,s}} \times t_{\mathrm{op},p,s} \times T_{p,s} \right) + \sum_{p=1}^{n_{\mathrm{pumps}}} \left(\mathrm{DC}_p \times \alpha_p \right), \end{aligned} \tag{8.3}$$

where n_{steps} is the number of time-steps in which the simulation is divided, n_{pumps} is the number of pumps to control, t_{op} is the pump operating time and $T_{p,s}$ is the price per energy unit defined for each pump p according to the tariff value for the time step s.

The second term of equation 8.3 corresponds to additional costs that can be included in periods of peak demand, which, in EPANET, is determined by the periods of pumping at maximum power. Such additional cost is computed by first searching, for all steps, the maximum value of power needed for each pump during the simulation ($P_{\max}$) and accounting the number of times it occurs. Then, the later value is multiplied by a demand charge (DC_p) defined by the user for each pump. Thus, α_p is given by the multiplication between the maximum power computed for a pump p, $P_{\max}$, and the number of times this value is repeated during the simulation, $n_{P_{\max}}$, *i.e.*

$$\alpha_p = P_{\max} \times n_{P_{\max}}. \tag{8.4}$$

Therefore, the first part of equation 8.3 corresponds to the electric energy consumption of pumps and the second part is related to additional costs related to the pumps when operating at maximum power.

As previously mentioned, the optimisation decision variables considered in the proposed methodology are defined by the relative pump speed M (fraction between the actual speed and the nominal speed, N_2/N_1) at each time-step of the simulation and also by the fraction of time of pump operation for the correspondent time-step as well as the fraction of time of the valves opening. Thus, the number of decision variables, n_{var}, can be determined by:

$$n_{\mathrm{var}} = \left(2n_{\mathrm{VSP}} + n_{\mathrm{FSP}} + n_{\mathrm{valves}}\right) n_{\mathrm{steps}}, \tag{8.5}$$

that represents two decision variables (speed and operating time) for each variable-speed pump of the network to be controlled at each time-step of the simulation and one variable (operating time) for each valve and/or fixed-speed pump of the network to be controlled also at each time-step. The number of variable-speed pumps, fixed-speed pumps, valves and time-steps are, respectively, n_{VSP}, n_{FSP}, n_{valves} and n_{steps}. Each variable can take values between 0 and 1, where 0 represents the minimum pump operating time or a pump relative speed inferior to the minimum allowed (user-defined value) and 1 represents the corresponding maximums.

In this work, a constraint to guarantee the continuity between the water levels in tanks in the beginning and in the end of simulation is considered. The equation that describes the continuity constraint is:

$$h_{1,i} = L_{i,\mathrm{final}} - L_{i,\mathrm{initial}} = 0, \qquad i = 1, \ldots, n_{\mathrm{tanks}}, \tag{8.6}$$

where $L_{i,\mathrm{final}}$ and $L_{i,\mathrm{initial}}$ are, respectively, the final and initial water level of each tank i and n_{tanks} is the number of tanks. Alternatively, and to simplify the optimisation problem*, this constraint may be

*Note that inequality constraints are easier to handle than equality constraints (Andrade-Campos, Dias-de-Oliveira, & Pinho-da-Cruz, 2015).

defined as an inequality:

$$g_{1,i} = L_{i,\text{final}} - L_{i,\text{initial}} \leq 0, \qquad i = 1, ..., n_{\text{tanks}}. \tag{8.7}$$

Naturally, a solution with lower costs associated will be obtained for differences between the initial and the final water levels close to zero.

Constraints for the maximum and minimum tank water levels are also applied. The constraint for the maximum water level allowed is given by:

$$g_{2,i} = L_i - L_{i,\text{max}} \leq 0, \qquad i = 1, ..., n_{\text{tanks}} \tag{8.8}$$

and for the minimum water level allowed by:

$$g_{3,i} = L_{i,\text{min}} - L_i \leq 0, \qquad i = 1, ..., n_{\text{tanks}} \tag{8.9}$$

where $L_{i,\text{max}}$ and $L_{i,\text{min}}$ are the maximum and the minimum water levels of operation for each tank i, respectively.

Concerning the nodal pressure constraints, the hydraulic simulator EPANET only controls minimum pressures by not allowing negative values. However, in water networks there are minimum (and sometimes maximum) values for pressure in certain nodes that need to be respected. Thus, similarly to the water levels in tanks, a constraint for the minimum pressure allowed is treated as:

$$g_{4,i} = P_{i,\text{min}} - P_i \leq 0, \qquad i = 1, ..., n_{\text{cnodes}} \tag{8.10}$$

where $P_{i,\text{min}}$ is the minimum pressures allowed in each i control node and n_{cnodes} represents the number of nodes that need to be controlled in terms of pressure. In case of violation of the described constraints, the objective function is penalised through the use of penalty methods (see Section 8.3).

Other constraints are defined by the hydraulic simulator through the set of equations for mass and energy conservation. During the evaluation of energy costs, the hydraulic simulator may converge to unbalanced/unstable results (in case it is not possible to converge to a hydraulic solution) or produce errors such as negative pressures or pumps operating outside the limits imposed by their curves. In order to ensure the proper system operation, variables that induce such errors are considered infeasible and cannot be accepted. For this reason, in the presented work, hydraulic unbalanced results are also treated as constraints, controlled by the addition of a penalty to the objective function in case of constraint violation. For any kind of unbalanced or unstable hydraulic situation (not possible to converge to a stable/balanced system operation or elements operating outside their limits), EPANET sends out a warning message. Thus, a warning-related constraint function was defined, constraining the number of warning messages (n_{warn}) resulting from the hydraulic analysis to zero (equality constraint):

$$h_2 = n_{\text{warn}} = 0. \tag{8.11}$$

In summary, the optimisation approach presented in this work can be mathematically described

by:

$$
\begin{aligned}
\underset{\mathbf{X}}{\text{minimise}} \quad & C_{\text{total}}(\mathbf{X}) = \gamma \sum_{p=1}^{n_{\text{pumps}}} \sum_{s=1}^{n_{\text{steps}}} \left[\frac{H_{p,s}(\mathbf{X}) Q_{p,s}(\mathbf{X})}{\eta_{p,s}(\mathbf{X})} t_{\text{op},p,s} T_{p,s} \right] + \sum_{p=1}^{n_{\text{pumps}}} \left[\text{DC}_p \alpha_p(\mathbf{X}) \right], \\
\text{subject to:} \quad & g_{1,j}(\mathbf{X}) = L_{j,\text{final}}(\mathbf{X}) - L_{j,\text{initial}}(\mathbf{X}) \leq 0, && j = 1, ..., n_{\text{tanks}}, \\
& g_{2,j}(\mathbf{X}) = L_j(\mathbf{X}) - L_{j,\max} \leq 0, && j = 1, ..., n_{\text{tanks}}, \\
& g_{3,j}(\mathbf{X}) = L_{j,\min} - L_j(\mathbf{X}) \leq 0, && j = 1, ..., n_{\text{tanks}}, \\
& g_{4,k}(\mathbf{X}) = P_{k,\min} - P_k(\mathbf{X}) \leq 0, && k = 1, ..., n_{\text{cnodes}}, \\
& h_2(\mathbf{X}) = n_{\text{warn}}(\mathbf{X}) = 0, \\
& 0 \leq x_i \leq 1, && i = 1, ..., n_{\text{var}},
\end{aligned}
\tag{8.12}
$$

where

$$
\begin{aligned}
\mathbf{X} &= \begin{bmatrix} x_{1,1} & \cdots & x_{n_{\text{comp}},1} \\ \vdots & \ddots & \vdots \\ x_{1,n_{\text{steps}}} & \cdots & x_{n_{\text{comp}},n_{\text{steps}}} \end{bmatrix}^{\text{T}} \\
&= \begin{bmatrix}
t_{\text{op},1,1}^{\text{VSP}} & \cdots & t_{\text{op},vs,1}^{\text{VSP}} & M_{1,1} & \cdots & M_{vs,1} & t_{\text{op},1,1}^{\text{FSP}} & \cdots & t_{\text{op},fs,1}^{\text{FSP}} & t_{\text{op},1,1}^{\text{Valve}} & \cdots & t_{\text{op},vl,1}^{\text{Valve}} \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
t_{\text{op},1,s}^{\text{VSP}} & \cdots & t_{\text{op},vs,s}^{\text{VSP}} & M_{1,s} & \cdots & M_{vs,s} & t_{\text{op},1,s}^{\text{FSP}} & \cdots & t_{\text{op},fs,s}^{\text{FSP}} & t_{\text{op},1,s}^{\text{Valve}} & \cdots & t_{\text{op},vl,s}^{\text{Valve}}
\end{bmatrix}^{\text{T}}
\end{aligned}
\tag{8.13}
$$

represents the matrix of the decision variables, with $n_{\text{comp}} = 2n_{\text{VSP}} + n_{\text{FSP}} + n_{\text{Valves}}$, $vs = n_{\text{VSP}}$, $fs = n_{\text{FSP}}$, $vl = n_{\text{valves}}$ and $s = n_{\text{steps}}$. The number of components, n_{com}, includes (i) the operating times for variable-speed pumps, fixed-speed pumps and valves that are represented by $t_{\text{op}}^{\text{VSP}}$, $t_{\text{op}}^{\text{FSP}}$ and $t_{\text{op}}^{\text{Valve}}$, respectively and (ii) the relative speed for variable-speed pumps, represented by M.

8.2 Optimisation methods

Optimisation processes can be seen in people day-to-day life and in nature. In nature, for instance, physical systems tend to states of minimum energy, the species evolve by knowledge transmission and by keeping the best genes through generations. In our day-to-day, optimisation is applied to a wide range of areas, maximising the efficiency of processes, minimising the amount of material in certain structures, finding the best routes, etc. Even in the execution of simple tasks (such as cooking, playing a game, etc.), people and animals learn with experience and tend to improve/optimise the tasks execution in the future. These last examples are a kind of *trial-and-error* optimisation techniques.

Nowadays, the industry requires the best results as fast as possible in almost all situations. The use of *trial-and-error* techniques are not efficient for these cases for being so time consuming and not cost-effective at all. For this reason, engineers often rely on computer programmes applying optimisation algorithms to solve the wide range of existing problems that appear in the real world.

8.2.1 Classification of non-linear optimisation methods

Dealing with real problems implies facing non-linear problems, characterised by non-linear objective functions and/or by non-linear constraint functions.

Non-linear optimisation methods can be characterised according to three main families: (i) the gradient-based methods that use approximations of the Hessians (Newton's method or Quasi-Newton methods) or the gradients of the objective function (Steepest Descent, Conjugate Gradient methods, etc.); (ii) the direct-search or derivative-free methods, considering only the function values, such as the Hill Climbing method, the Powell's method (or conjugate directions method) or the Nelder-Mead Simplex method; and (iii) the nature-inspired methods, including the evolutionary algorithms (Genetic Algorithms, Differential Evolution, etc.), the algorithms based on collective knowledge of groups (Ant Colony Optimisation, Particle Swarm Optimisation, etc.) or even the artificial-intelligence-based algorithms (Fuzzy Systems, Artificial Neural Networks, etc.). However, optimisation techniques are often classified in several distinct ways. Rao (2009), for example, classify the optimisation methods according to two main families: (i) the classical or traditional methods and (ii) the modern or non-traditional methods, distinguishing essentially the mathematical programming techniques from the modern ones typically inspired on nature. In turn, Nocedal and Wright (2006) distinguishes the algorithms according to the nature of the optimisation problem: (i) constrained or (ii) unconstrained optimisation. Yang (2010) classifies the optimisation algorithms as (i) deterministic, including the most conventional or classic algorithms such as non-linear programming, gradient-based, gradient-free, and (ii) stochastic, referring to heuristic and/or meta-heuristic algorithms from the population-based to the trajectory-based ones.

The non-linear optimization methods based on the gradient function can find local minima through an iterative process whatever the initial guess of solution. However, frequently these are not the absolute minimum values. In major situations, the solution also depends on the initial parameters, leading to different final results (Valente, Andrade-Campos, Carvalho, & Cruz, 2011). The modern methods based on nature have the advantage of not requiring the function derivatives revealing a large flexibility in modelling engineering problems but, at the same time, present as main disadvantage the typically larger computational times (Costa, 2003). These kinds of algorithms are probabilistic. Consequently, these do not depend on the initial variables and are probabilistic capable of finding the global minimum.

In the following subsections, a brief description concerning the optimisation algorithms addressed in this work is provided.

8.2.2 Nelder-Mead Simplex - NMSimplex

The Nelder-Mead Simplex (NMSimplex) method, also known as downhill simplex[†], is a derivative-free pattern-search algorithm for unconstrained optimisation (Yang, 2010) and was developed by J. A. Nelder and R. Mead in 1965 (Nelder & Mead, 1965).

A simplex is a geometric figure formed by a set of $v+1$ vertices in a v-dimensional space[‡]. In the iterative optimisation procedure, the NMSimplex algorithm compares the values of the objective

[†]This method is called simplex due to the use of a geometrical simplex. It is not related with the simplex method used in linear programming

[‡]Considering two variables, *i.e.* a two dimensional space, the simplex is a triangle.

function at the $v+1$ vertices and gradually move the simplex toward the optimum point (Rao, 2009). In other words, the idea is to remove the vertex with the worst function value and replace it by a point with a better value, obtained by reflecting, expanding or contracting the simplex. These movements of the simplex are performed along the line joining the worst vertex with the centroid of the remaining vertices. If no better point is found, the vertex with the best value is maintained and the remaining vertices are moved toward such value (Nocedal & Wright, 2006). This is an effective and computationally compact algorithm (Nelder & Mead, 1965).

The method starts by (i) choosing an initial simplex (by picking a random location, for example), (ii) taking unit vectors for the edges, (iii) evaluating the objective function values at each vertex and (iv) ordering such values (Nocedal & Wright, 2006; Walski et al., 2001; Yang, 2010):

$$f(\mathbf{x}_1) \leq f(\mathbf{x}_2) \leq ... \leq f(\mathbf{x}_{v+1}), \tag{8.14}$$

with $f(\mathbf{x}_1)$ representing the best value and $f(\mathbf{x}_{v+1})$ the worst, considering a minimisation problem. After that, the centroid (mean) of the simplex, $\mathbf{x}_0$, is computed excluding the worst vertex:

$$\mathbf{x}_0 = \frac{1}{v}\sum_{i=1}^{v}\mathbf{x}_i. \tag{8.15}$$

After obtaining the centroid, the operations for moving the simplex are performed. First, the reflected point $\mathbf{x}_\mathrm{r}$ is computed by:

$$\mathbf{x}_\mathrm{r} = (1+\alpha_\mathrm{r})\mathbf{x}_0 - \alpha_\mathrm{r}\mathbf{x}_{v+1}, \qquad \text{with} \qquad \alpha_\mathrm{r} > 0, \tag{8.16}$$

where α_r is the reflection coefficient and can be defined by (Rao, 2009):

$$\alpha_\mathrm{r} = \frac{\text{distance between } \mathbf{x}_\mathrm{r} \text{ and } \mathbf{x}_0}{\text{distance between } \mathbf{x}_{v+1} \text{ and } \mathbf{x}_0}. \tag{8.17}$$

The objective function is then evaluated at $\mathbf{x}_\mathrm{r}$ and one of the three possibilities can occur:

- If the reflected point is not the best neither the worst of the simplex, then the worst point, $\mathbf{x}_{v+1}$, gets its value and a new iteration starts;
- If the reflected point is better than the current best, then continue in this direction, expanding the vertex, trying to improve further:
  - The expanded point, $\mathbf{x}_\mathrm{e}$, is computed by

    $$\mathbf{x}_\mathrm{e} = \gamma_\mathrm{e}\mathbf{x}_\mathrm{r} + (1-\gamma_\mathrm{e})\mathbf{x}_0, \qquad \text{with} \qquad \gamma_\mathrm{e} > 1, \tag{8.18}$$

    where γ_e is the expansion coefficient and can be defined by (Rao, 2009):

    $$\gamma_\mathrm{e} = \frac{\text{distance between } \mathbf{x}_\mathrm{e} \text{ and } \mathbf{x}_0}{\text{distance between } \mathbf{x}_\mathrm{r} \text{ and } \mathbf{x}_0}. \tag{8.19}$$

    The objective function is then evaluated at $\mathbf{x}_\mathrm{e}$.
  - If the expanded point is better than the reflected point, then it replaces it and a new iteration starts.

  - Otherwise, the reflection point replaces the worst point, $\mathbf{x}_{\nu+1}$, and a new iteration starts.

- If the reflected point is the worst of the simplex, *i.e.* there is no improvement, then the size of the simplex is reduced by contraction:

  - The contracted point, $\mathbf{x}_c$, is computed by:

$$\mathbf{x}_c = \beta_c \mathbf{x}_{\nu+1} + (1-\beta_c)\mathbf{x}_0, \qquad \text{with} \qquad 0 \leq \beta_c \leq 1, \tag{8.20}$$

    where β_c represents the contraction coefficient (usually $\beta_c = 1/2$) and can be determined by:

$$\beta_c = \frac{\text{distance between } \mathbf{x}_c \text{ and } \mathbf{x}_0}{\text{distance between } \mathbf{x}_{\nu+1} \text{ and } \mathbf{x}_0}. \tag{8.21}$$

    The objective function is then evaluated at $\mathbf{x}_c$.

  - If the contracted point improves the worst ($\mathbf{x}_{\nu+1}$), then it replaces the worst and a new iteration starts;

  - Otherwise, the simplex is reduced towards the best vertex $\mathbf{x}_1$ (shrink operation):

$$\mathbf{x}_i = \delta \mathbf{x}_i + (1-\delta_s)\mathbf{x}_1, \qquad \text{for} \qquad i = 2, ..., \nu+1, \tag{8.22}$$

    i.e. all the points, except the best, are replaced and then a new iteration starts. The coefficient δ_s is usually called the shrink coefficient.

Standard values for the operations coefficients are $\alpha_r = 1$, $\gamma_e = 2$, $\beta_c = 1/2$ and $\delta_s = 1/2$.

The stop criteria for this method can be defined according to a maximum number of iterations or when the convergence criteria is reached, *i.e.* (Rao, 2009),

$$C_{\text{stop}} = \sqrt{\frac{1}{\nu+1}\sum_{i=1}^{\nu+1}\left[f(\mathbf{x}_i) - f(\mathbf{x}_0)\right]^2} \leq \varepsilon_{\text{stop}}, \tag{8.23}$$

which states that the standard deviation of the function at the $\nu+1$ vertices is smaller or equal to an user-defined small quantity $\varepsilon_{\text{stop}}$ (*e.g.* $\varepsilon_{\text{stop}} = 0.2$).

The pseudo-code for the implementation of the Nelder-Mead Simplex is provided in Algorithm 1. Figure 8.1 provides examples of the previously described operations in a simplex generated by 2 variables (triangle).

8.2.3 Genetic Algorithms - GA

Genetic Algorithms (GA) are adaptive heuristic search algorithms based on the evolutionary ideas of natural selection and genetics. The theory behind GA was firstly introduced by John Holland in the early seventies and further developed by Goldberg and other collaborators (Walski et al., 2001).

The basic techniques applied in GA attempt to simulate processes of natural evolution, sustained by the evolutionary theory of Charles Darwin of survival of the fittest individuals through generations by competing and dominating the weakest individuals. Individuals with better fitness have large probability to survive, to attract mates and thus, to propagate their genetic information through generations.

Algorithm 1 Pseudo-code for Nelder-Mead Simplex

Initialise a simplex with $\nu+1$ vertices in ν dimension and operation algorithm coefficients;
while $C_{\text{stop}} > \varepsilon_{\text{stop}}$ **do**
  Order the points (solution) such that $f(\mathbf{x}_1) \leq f(\mathbf{x}_2) \leq ... \leq f(\mathbf{x}_{\nu+1})$, where $\mathbf{x}_1$ represents the best and $\mathbf{x}_{\nu+1}$ the worst solution;
  Find the centroid, $\mathbf{x}_0 = \sum_{i=1}^{\nu} \mathbf{x}_i/\nu$, excluding $\mathbf{x}_{\nu+1}$;
  Generate a reflected point $\mathbf{x}_{\text{r}} = (1+\alpha_{\text{r}})\mathbf{x}_0 - \alpha_{\text{r}}\mathbf{x}_{\nu+1}$;
  if $f(\mathbf{x}_1) < f(\mathbf{x}_{\text{r}}) < f(\mathbf{x}_{\nu+1})$ **then**
    $\mathbf{x}_{\nu+1} \leftarrow \mathbf{x}_{\text{r}}$;
    go to step 3.
  else
    if $f(\mathbf{x}_{\text{r}}) < f(\mathbf{x}_1)$ **then**
      Expand in the direction of reflection $\mathbf{x}_{\text{e}} = \gamma_{\text{e}}\mathbf{x}_{\text{r}} + (1-\gamma_{\text{e}})\mathbf{x}_0$;
      if $f(\mathbf{x}_{\text{e}}) < f(\mathbf{x}_{\text{r}})$ **then**
        $\mathbf{x}_{\nu+1} \leftarrow \mathbf{x}_{\text{e}}$;
        go to step 3.
      else
        $\mathbf{x}_{\nu+1} \leftarrow \mathbf{x}_{\text{r}}$;
        got to step 3.
      end if
    else
      if $f(\mathbf{x}_{\text{r}}) > f(\mathbf{x}_{\nu+1})$ **then**
        Contract by $\mathbf{x}_{\text{c}} = \beta_{\text{c}}\mathbf{x}_{\nu+1} + (1-\beta_{\text{c}})\mathbf{x}_0$;
        if $f(\mathbf{x}_{\text{c}}) < f(\mathbf{x}_{\nu+1})$ **then**
          $\mathbf{x}_{\nu+1} \leftarrow \mathbf{x}_{\text{c}}$;
          go to step 3.
        else
          Reduce by $\mathbf{x}_i = \delta_{\text{s}}\mathbf{x}_i + (1-\delta_{\text{s}})\mathbf{x}_1$, with $i = 2, ..., \nu+1$;
          go to step 3.
        end if
      end if
    end if
  end if
end while
return Best solution.

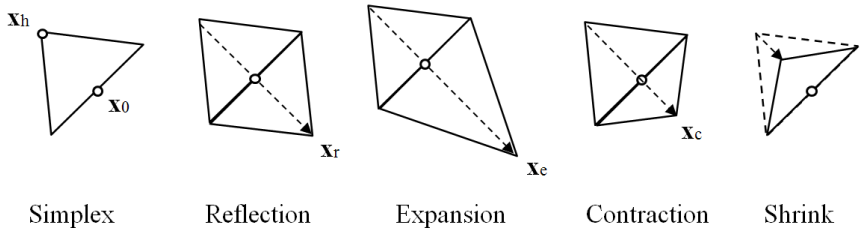


Figure 8.1: Operations performed in the Nelder-Mead Simplex algorithm considering a simplex in a two-dimensional space. $\mathbf{x}_{\text{h}}$ represents the highest (worst) point, $\mathbf{x}_0$ the centroid (mean) and $\mathbf{x}_{\text{r}}$, $\mathbf{x}_{\text{e}}$ and $\mathbf{x}_{\text{c}}$ represent, respectively, reflected, expanded (after reflection) and contracted (also after reflection) points.

The main steps followed by a GA can be simply described as (Walski et al., 2001):

1. Random generation of an initial population of solutions;
2. Fitness computation (evaluation of the objective function) of each solution in the initial population;
3. Generation of a new population using biologically inspired operators (selection, crossover and mutation);
4. Fitness computation of the new solutions;
5. Generations evolution[§] by repeating the steps 3 to 5 until find the desired conditions (stop criteria).

This algorithm starts not with a single initial solution but a set of possible solutions (that can be randomly generated), called population of individuals. Each individual (solution) is represented by a chromosome that, in turn, is constituted by genes (variables). In a standard GA, the chromosomes are encoded with binary numbers. However, sequences of real numbers, characters or objects can also be used for encoding the variables of a solution. In a problem with n_{var} variables, the population size is typically set to $2n_{\text{var}}$ to $4n_{\text{var}}$ (Rao, 2009).

The evaluation of the objective function for a certain solution gives the *fitness*, a measure of how good the individual (solution) is at competition. In case of minimisation problems, the best fitness corresponds to the lowest value of the objective function.

Since the generation of new species occurs from sexual reproduction, in the GA algorithm similar processes of individuals' selection and combination of the genetic material (chromosomes) for the generation of new individuals are performed.

The selection of individuals is performed using a *selection* operator, based on a probabilistic procedures to pick individuals from the current population. Usually, the probability of selection of an individual is proportional to its fitness (Rao, 2009). The idea is to give preference to better individuals, allowing them to pass on their genes to the next generation. Selection procedures commonly used are (Goldberg & Deb, 1991):

Proportionate selection, also known as *roulette wheel selection* for being similar with a roulette wheel in a casino. Parents are probabilistically selected according to their fitness. The better the chromosomes are, the more chances to be selected they have. Imagining a roulette wheel where all chromosomes in the population are placed, every chromosome has its place with size proportional to its fitness function. Then a random selection is made similar to how the roulette wheel is rotated. The main drawback of this selection method is that some chromosomes may have few chances of being selected.

Ranking selection, a method that ranks the population and then every chromosome receives fitness from this ranking. The worst will have fitness 1, second worst 2, etc., and the best will have fitness n_{p} (number of chromosomes in population). Consequently, all the chromosomes have a chance to be selected. The main disadvantage is that the best chromosomes do not differ so much from other ones, which can lead to slower convergence.

Tournament selection, as the name says, the method involves running several "tournaments" among a few individuals randomly chosen from the population. The winner of each tournament (the one with the best fitness) is selected for crossover. The tournament size can be adjusted. If the size is larger, weak individuals have a smaller chance to be selected.

[§]Each generation corresponds to one iteration of the algorithm.

The recombination operator, usually called *crossover*, takes two selected individuals (parents) and cuts their chromosome at a random point (or points). The new individuals (offspring) are obtained by recombining portions of the parents chromosomes (see crossover example in Figure 8.2).

The introduction of random modifications in the chromosomes is performed by the *mutation* operator, which replaces one or more specific genes in the new individuals (see mutation example in Figure 8.2).

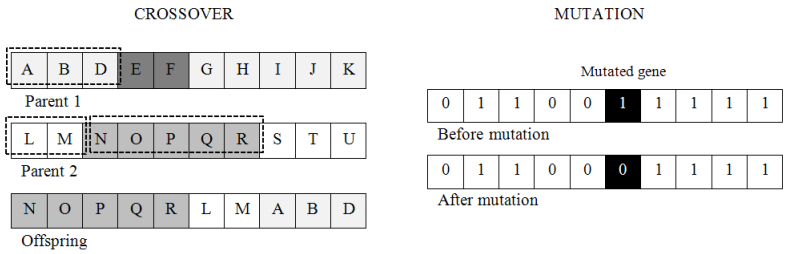


Figure 8.2: Examples of operations that can occur between chromosomes in a Genetic Algorithm. In the 2-points crossover operation, each parent is crossed over in 2 points. The parts recombined to generate the offspring are signed by the dashed lines. In the mutation operation, only one gene is replaced by a new one in order to generate a distinct chromosome. For demonstration, a distinct type of variables codification is presented for each example (characters and binary strings).

The last two operations do not occur to all the individuals in the population. A probability of occurrence is initially pre-defined. While the considered crossover probability, or rate of crossover, r_c, is typically high, in the range of $0.7-1.0$ (70 to 100 % of probability), the mutation probability, r_m, is usually small, in the range of $0.001-0.05$. This is because high probability of mutation may cause too large "jumps" in the search space even if the optimal solution is getting closed (Yang, 2010).

The pseudo-code for the implementation of a typical GA is provided in Algorithm 2.

8.2.4 Differential Evolution - DE

Differential Evolution (DE) is also an evolutionary algorithm and was initially introduced by Storn and Price (1995). As the GA, DE is also based on the natural evolution of species. However, in this algorithm, the operations used for the population perturbation consist in computing differences between vectors.

In DE algorithm, the individuals (candidate solutions) are moved around in the search-space by using simple mathematical formulae to combine the positions of the existing individuals in the population. If the new position of an individual is an improvement (*i.e.* the value of objective function is reduced, for a minimisation problem) it is accepted and forms part of the population, otherwise the new position is simply discarded.

For an optimisation problem with n decision variables, each individual (parameter vector) of a DE population takes the form:

$$\mathbf{x}_i^G = [x_{i,1}^G, x_{i,2}^G, ..., x_{i,v}^G] \qquad \text{with} \qquad i = 1, 2, ..., n_P, \tag{8.24}$$

where G is the generation number and n_P is the population size. Thus, the n_P individuals of the initial

Algorithm 2 Pseudo-code for Genetic Algorithm

```
Define population size ($n_P$), number of generations ($n_G$), rate of crossover ($r_c$) and rate of mutation ($r_m$);
Initialise population with $n_P$ individuals;
Evaluate population according to fitness criteria;
while stopping criteria (typically $n_G$ iterations) is not verified do
    Create new solutions:
    for Each two of $n_P$ individuals (solutions) do
        Select two parent solutions from the current population;
        if rand[0,1] < $r_c$ then
            Create two offspring solutions using crossover;
        else
            Set parents as new solutions;
        end if
        if rand[0,1] < $r_m$ then
            Mutate the new solutions;
        end if
        Evaluate the new solutions and rank them by fitness. The best are kept.
    end for
end while
return Best solution.
```

population ($G = 0$) can be randomly generated in the ν-dimensional search-space by:

$$x_{i,j}^0 = \text{rand}[0,1](x_j^{\max} - x_j^{\min}) + x_j^{\min}, \qquad \text{with} \qquad j = 1,2,...,\nu, \tag{8.25}$$

considering the variables boundaries:

$$x_j^{\min} \leq x_j \leq x_j^{\max}, \tag{8.26}$$

where $x_j^{\min}$ and $x_j^{\max}$ are, respectively, the lower and upper bounds for the j variable, and rand$[0,1]$ is a randomly generated number between 0 and 1. Consecutively, the DE algorithm performs the mutation operation in which, for each target vector $\mathbf{x}_i^G$, a disturbed vector $\mathbf{x}_i^{G+1}$ is generated by adding a weighted difference to a third vector, according to the equation (Storn & Price, 1995):

$$\mathbf{x}_i^{G+1} = \mathbf{x}_{r_1}^G + F(\mathbf{x}_{r_2}^G - \mathbf{x}_{r_3}^G), \tag{8.27}$$

where F is a weight coefficient, usually called mutation factor, and can take positive real numbers in the range of $[0,2]$. This coefficient controls the amplification of the differential variation. r_2 and r_3 are random integers from $1,2,...,n_P$, but different from i. Depending on the mutation strategy, r_1 can either be a random integer or the best member of the population. Figure 8.3 demonstrates how a mutated vector is generated in a two-dimensional function.

After applying the mutation operator, the crossover operation is performed, allowing the incorporation of successful solutions from the previous generation. A trial vector $\mathbf{u}_i^{G+1}$ is generated from the elements of the target vector $\mathbf{x}_i^G$ and the elements of the disturbed (mutated) vector $\mathbf{x}_i^{G+1}$. A fraction of elements j of the disturbed vector are copied to this trial vector with a crossover probability CR,

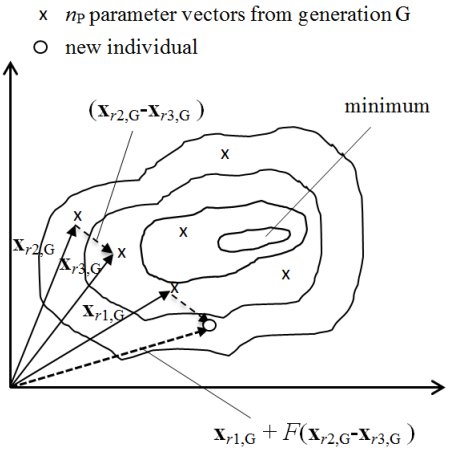


Figure 8.3: Example showing the process in DE algorithm for generating a mutated vector for the minimisation of a two-dimensional function.

according to:

$$u_{i,j}^{G+1} = \begin{cases} x_{i,j}^{G} & \text{if } \text{rand}_{i,j}[0,1] \leq \text{CR or } j = \text{I}_{\text{rand}} \in \{1,...,\nu\} \\ x_{i,j}^{G+1} & \text{if } \text{rand}_{i,j}[0,1] > \text{CR and } j \neq \text{I}_{\text{rand}} \in \{1,...,\nu\}, \end{cases} \tag{8.28}$$

where I_{rand} is a random integer picked from $1,...,\nu$, which ensures that $x_{i,j}^{G} \neq x_{i,j}^{G+1}$. The crossover probability, can take values in the range of $[0,1]$. This crossover process is similar to the GA crossover.

Finally, the trial vector $\mathbf{u}_i^{G+1}$ is compared with the target $\mathbf{x}_i^{G}$ for the following selection process:

$$\mathbf{x}_i^{G+1} = \begin{cases} \mathbf{u}_i^{G+1} & \text{if } f(\mathbf{u}_i^{G+1}) \leq f(\mathbf{x}_i^{G}) \\ \mathbf{x}_i^{G} & \text{otherwise,} \end{cases} \qquad \text{with} \qquad i = 1,...,n_{\text{P}}. \tag{8.29}$$

If $\mathbf{u}_i^{G+1}$ has a better fitness, it replaces $\mathbf{x}_i^{G}$ in the $G+1$ generation. Otherwise $\mathbf{x}_i^{G}$ is maintained. These processes, including mutation, crossover and selection, is repeated until a stopping criteria is satisfied.

Algorithm 3 provides a pseudo-code for the implementation of a simple variant of DE.

8.2.5 Particle Swarm Optimisation - PSO

Particle swarm optimisation (PSO) is a population-based meta-heuristic optimisation technique developed by Kennedy and Eberhart (1995) and inspired in social behaviour, simulating the movement of organisms such as a colony of bees, a flock of birds or a shoal of fish.

PSO shares similarities with evolutionary computation techniques such as GA and DE. The PSO algorithm is initialised with a population of (random) solutions and searches for the optimum by their updating in each iteration. However, unlike GA and DE (evolution-based algorithms), PSO has no evolution operators such as crossover or mutation. In PSO, the potential solutions, called particles, fly through the search-space by following the current best particles and cooperation.

A basic variant of the PSO algorithm works by having a population (called *swarm*) of candidate solutions (called *particles*). The particles of the swarm are moved around in the search-space according to a few simple formulae. The movements of the particles are guided by their tracked information about themselves and the neighbourhood. When improved positions are discovered between the iter-

Algorithm 3 Pseudo-code for Differential Evolution

Define population size (n_{P}), number of generations (n_{G}), crossover probability (CR) and mutation factor (F);
Initialise all n_{P} individuals with random positions in the search-space. $G = 0$.
while stopping criteria is not verified **do**
  for Each individual $\mathbf{x}_i^G$ (with $i = 1, ..., n_{\mathrm{P}}$) from the population **do**
    Randomly pick three candidate solutions $\mathbf{x}_{r_1}^G$, $\mathbf{x}_{r_2}^G$, and $\mathbf{x}_{r_3}^G$ from the population. They must be distinct from each other as well as from $\mathbf{x}_i^G$;
    Pick a random integer $\mathrm{I}_{\mathrm{rand}} \in \{1, \ldots, \nu\}$;
    Disturb the target vector $\mathbf{x}_i^G$ using mutation operation: $\mathbf{x}_i^{G+1} = \mathbf{x}_{r_1}^G + F(\mathbf{x}_{r_2}^G - \mathbf{x}_{r_3}^G)$;
    Obtain a trial vector (offspring), $\mathbf{u}_i^{G+1}$, using crossover operation according to the probability CR:
    for each element j **do**
      if $\mathrm{rand}_{i,j}[0,1] \leq \mathrm{CR}$ or $j = \mathrm{I}_{\mathrm{rand}} \in \{1, ..., \nu\}$ **then**
        $u_{i,j}^{G+1} = x_{i,j}^G$
      else
        $u_{i,j}^{G+1} = x_{i,j}^{G+1}$
      end if
    end for
    Evaluate the fitness, f, and perform the selection operation:
    if $f(\mathbf{u}_i^{G+1}) \leq f(\mathbf{x}_i^G)$ **then**
      replace the target individual, $\mathbf{x}_i^G$, by the improved candidate solution, $\mathbf{u}_i^{G+1}$.
    end if
  end for
  $G = G + 1$;
end while
return Best solution.

ative cycles, such information is used to guide the movements of the swarm in the next iteration. The process is repeated and, by doing so, it is hoped, but not guaranteed, that a satisfactory solution will eventually be discovered.

In this algorithm, the initial particles are randomly set in the search-space. A vector position, $\mathbf{x}_i$, and a vector velocity, $\mathbf{v}_i$, are associated to each particle i. In each iteration $t = 1, ..., n_{\mathrm{iter}}$, the velocity of each particle, $\mathbf{v}_i^t$, is updated according to (Rao, 2009):

$$\mathbf{v}_i^t = \mathbf{v}_i^{t-1} + c_1 \mathrm{rand}[0,1] \left[\mathbf{P}_i^{\mathrm{best}} - \mathbf{x}_i^{t-1}\right] + c_2 \mathrm{rand}[0,1] \left[\mathbf{G}^{\mathrm{best}} - \mathbf{x}_i^{t-1}\right], \tag{8.30}$$

where c_1 and c_2 are, respectively, the cognitive (individual) and the social (group) learning rates that provide the relative importance that each particle gives to its own information and to its neighbours information when performing the next step. $\mathbf{P}_i^{\mathrm{best}}$ and $\mathbf{G}^{\mathrm{best}}$ are, respectively, the particle best position and the global best position found by the swarm up to the current iteration.

After computing the velocity vectors, the position of each particle can be updated taking into account such velocities by (Rao, 2009):

$$\mathbf{x}_i^t = \mathbf{x}_i^{t-1} + \mathbf{v}_i^t, \tag{8.31}$$

where $\mathbf{x}_i^{t-1}$ represents the particle position in the previous iteration $t - 1$.

The new position computed for each particle is then compared with the particle's own best and with the global best of the swarm.

A pseudo-code describing the implementation of a basic variant of PSO is provided in Algorithm 4.

Algorithm 4 Pseudo-code for Particle Swarm Optimisation

Define the swarm size (n_S), the number of iterations (n_{iter}), the cognitive learning rate (c_1) and the social learning rate (c_2);
Initialise all $i = 1, ..., n_S$ particles with random initial positions, $\mathbf{x}_i$, and a initial velocity, $\mathbf{v}_i$, in the search-space (typically all initial velocities are assumed to be zero);
Evaluate all particles and track the current global best position of the swarm, $\mathbf{G}^{\text{best}}$;
Initialise the particle's current best known position: $\mathbf{P}_i^{\text{best}} \leftarrow \mathbf{x}_i$;
while stopping criteria is not verified **do**
    for each particle i in the iteration t **do**
        Find current particle velocity by:
        $\mathbf{v}_i^t = \mathbf{v}_i^{t-1} + c_1 \text{rand}[0,1]\left[\mathbf{P}_i^{\text{best}} - \mathbf{x}_i^{t-1}\right] + c_2 \text{rand}[0,1]\left[\mathbf{G}^{\text{best}} - \mathbf{x}_i^{t-1}\right]$;
        Find current particle position by :
        $\mathbf{x}_i^t = \mathbf{x}_i^{t-1} + \mathbf{v}_i^t$
        Update particle historical best:
        if $f(\mathbf{x}_i^t) < f(\mathbf{P}_i^{\text{best}})$ **then**
            $\mathbf{P}_i^{\text{best}} \leftarrow \mathbf{x}_i^t$
        end if
        Update current global best of the swarm:
        if $f(\mathbf{P}_i^{\text{best}}) < f(\mathbf{G}^{\text{best}})$ **then**
            $\mathbf{G}^{\text{best}} \leftarrow \mathbf{P}_i^{\text{best}}$;
        end if
    end for
    $t = t + 1$;
end while
return best particle.

8.2.6 Simulated Annealing - SA

The Simulated Annealing (SA) method is based on the cooling process of molten metals by annealing. This method was developed by Kirkpatrick, Gelatt, and Vecchi (1983).

Since the fast cooling may introduce defects in the material, the temperature of the molten metal during the cooling process should be controlled and slowly reduced. This controlled process called annealing ensures the proper solidification of the material with an high order of crystalline state that corresponds to the minimum energy states of the particles. In order to simulate this kind of process, the SA algorithm applies a temperature parameter, T_c, that is controlled according to the Boltzmann probability distribution (Rao, 2009):

$$P_{\text{trans}}(E_{\text{level}}) = \exp^{-E_{\text{level}}/k_B T_c}, \tag{8.32}$$

where $P_{\text{trans}}(E_{\text{level}})$ represents the probability of achieving the energy level E (also called transition probability) and k_B is the Boltzmann constant (typically, $k_B = 1$ is used).

The optimisation process with SA starts then with an initial guess solution (initial state) at a high temperature and gradually cools down the system until the state of minimum energy (best solution). The moves from one state of energy to other are performed (or not) according to some probability. The system should be cooled slowly enough in order to reach the global minimum (Yang, 2010).

Considering the current initial state (solution) $\mathbf{x}_t$, the energy $E_{\text{level},t}$ at such state is given by the value of the objective function:

$$E_{\text{level},t} = f(\mathbf{x}_t). \tag{8.33}$$

Then, the probability of the next state, $\mathbf{x}_{t+1}$, is given by the energy difference between the two consecutive states:

$$\Delta E_{\text{level}} = E_{\text{level},t+1} - E_{\text{level},t} = f(\mathbf{x}_{t+1}) - f(\mathbf{x}_t). \tag{8.34}$$

Thus, using the Boltzmann probability distribution, the new state (solution) can be found:

$$P_{\text{trans}}(E_{\text{level},t+1}) = \min\{1, \exp^{-\Delta E_{\text{level}}/k_{\text{B}}T_{\text{c}}}\}. \tag{8.35}$$

For $t = 1, ..., n_{\text{iter}}$ iterations, the previous process is repeated. After that, the cooling temperature, T_{c}, is replaced by $f_{\text{c}}T_{\text{c}}$, where $0 < f_{\text{c}} < 1$ is a cooling factor. This is called a geometric cooling schedule.

This trajectory-based search method is, in fact, a special case of a Markov chain by random walk (see Yang (2010) for more detail on this). Thus, global convergence is expected (and almost guaranteed) due to the stationary property of Markov chains. In turn, the main drawback of a trajectory-based algorithm is related with the expected slow convergence for complex optimisation problems (Yang, 2010).

A pseudo-code for the implementation of a simple variant of Simulated Annealing is provided in Algorithm 5.

Algorithm 5 Pseudo-code for Simulated Annealing

Define an initial state (solution) $\mathbf{x}_t$, a high initial temperature T_{c} and the number of iterations at a given temperature n_{iter};
while stopping criteria is not verified **do**
    for each $t = 1, ..., n_{\text{iter}}$ iteration **do**
        Pick a random new state in the neighbourhood $\mathbf{x}_{t+1}$;
        Compute the energy difference between the two states $\Delta E_{\text{level}} = f(\mathbf{x}_{t+1}) - f(\mathbf{x}_t)$
        if $\Delta E_{\text{level}} < 0$ **then**
            move to the new state;
        else
            if $P_{\text{trans}}(E_{\text{level},t+1}) > \text{rand}[0,1]$ **then**
                move to the new state;
            end if
        end if
    end for
    Reduce the temperature: $T_{\text{c}} \leftarrow f_{\text{c}}T_{\text{c}}$;
end while
return final (minimum) state of energy.

8.3 Constraint-handling methods

All the previously presented optimisation methods can be applied to unconstrained optimisation problems. However, several problems, including the operational control of water networks, are commonly subject to a number of constraints. To handle such constraints, two main approaches can be followed (Andrade-Campos et al., 2015; Rao, 2009; Yang, 2010): (i) adapting the unconstrained optimisation algorithm for constrained optimisation problems, in such a way the constraints are handled in a explicit manner (direct methods) or, alternatively, (ii) transforming the optimisation problem into an unconstrained problem (or sequential set of unconstrained problems) by using, for instance, penalty methods, Lagrange multipliers or techniques of variables transformation (indirect methods). While the first approach requires specific adaptations to each distinct optimisation algorithm, the second one allows an easier implementation of several distinct algorithms.

Only penalty methods are covered in this work since they are easy to implement and frequently used in non-linear constrained optimisation (Rao, 2009; Smith & Coit, 1997).

8.3.1 Penalty methods

There are two main types of penalty methods (Smith & Coit, 1997): (i) exterior penalty methods, which penalise infeasible solutions, and (ii) interior penalty methods, which penalise feasible solutions near the active constraint boundary. In both cases, the penalties can also be defined as static, dynamic or adaptive.

In the penalty methods, the constraints are represented by the addition of terms (also called penalty functions) to the objective function, each one providing the degree of violation of the corresponding constraint.

Considering the formulation of an optimisation problem, as the represented in Equation 8.2, and using a penalty method to solve such problem, it becomes:

$$\begin{aligned} \underset{\mathbf{x}}{\text{minimise}} \quad & f_{\mathrm{p}}(\mathbf{x}, r_{\mathrm{h},k}, r_{\mathrm{g},j}), \\ \text{subject to:} \quad & x_i^{\min} \leq x_i \leq x_i^{\max}, \qquad i = 1, \ldots, n_{\mathrm{var}}, \end{aligned} \tag{8.36}$$

where the objective function $f(\mathbf{x})$ is replaced by the penalised (transformed) function $f_{\mathrm{p}}(\mathbf{x}, r_{\mathrm{h},k}, r_{\mathrm{g},j})$. $r_{\mathrm{h},k}$ and $r_{\mathrm{g},j}$ are, respectively, the penalty coefficients for the equality and inequality constraints.

The penalised objective function, f_{p}, can be generally defined by (Rao, 2009):

$$f_{\mathrm{p}}(\mathbf{x}, r_{\mathrm{h},k}, r_{\mathrm{g},j}) = f(\mathbf{x}) + \sum_{k=1}^{K} F_k^{\mathrm{H}}\left(h_k(\mathbf{x}), r_{\mathrm{h},k}\right) + \sum_{j=1}^{J} F_j^{\mathrm{G}}\left(g_j(\mathbf{x}), r_{\mathrm{g},j}\right), \tag{8.37}$$

where F_j^{G} is a function of the inequality constraint g_j and the penalty coefficient for such constraint $r_{\mathrm{g},j}$. F_k^{H} is a function of the equality constraint $h_k(\mathbf{x})$ and the corresponding penalty coefficients $r_{\mathrm{h},k}$.

Figure 8.4 shows an example of the application of exterior and interior penalty methods to find the optimum of a function subject to a constraint whose boundary is represented in the figure (feasible and infeasible regions are represented by the gray areas).

For **interior penalty methods**, F_j^{G} should tend to the infinity as the constraint boundary is approached and F_k^{H} should also tend to the infinity and the other part of the corresponding term must

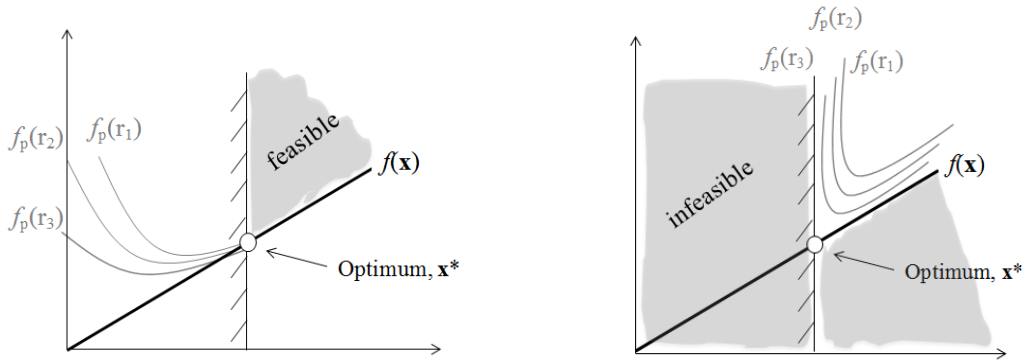


Figure 8.4: Demonstration of (a) exterior and (b) interior penalty methods considering distinct penalty coefficients r_1, r_2 and r_3 (with $r_1 > r_2 > r_3$).

tend to zero. The most common forms used for F_j^{G} are (Rao, 2009):

$$F_j^{\mathrm{G}} = -\frac{r_{\mathrm{g},j}}{g_j(\mathbf{x})}, \tag{8.38}$$

similarly with the represented in Figure 8.4b, and

$$F_j^{\mathrm{G}} = r_{\mathrm{g},j} \log[-g_j(\mathbf{x})]. \tag{8.39}$$

An example of a possible form for F_k^{H} is (Rao, 2009):

$$F_k^{\mathrm{H}} = \frac{[h_k(\mathbf{x})]^2}{\sqrt{r_{h,k}}}. \tag{8.40}$$

For **exterior penalty methods**, it is common to consider

$$F_j^{\mathrm{G}}(g_j(\mathbf{x}), r_{\mathrm{g},j}) = r_{\mathrm{g},j} \left[\max\{0; g_j(\mathbf{x})\}\right]^{\theta_{\mathrm{g}}} \tag{8.41}$$

and

$$F_k^{\mathrm{H}}(h_k(\mathbf{x}), r_{h,k}) = r_{h,k} \left[h_k(\mathbf{x})\right]^2, \tag{8.42}$$

where

$$\max\{0; g_j(\mathbf{x})\} = \begin{cases} g_j(\mathbf{x}), & \text{if } g_j(\mathbf{x}) > 0 \text{ (constraint is violated)}, \\ 0, & \text{if } g_j(\mathbf{x}) \leq 0 \text{ (constraint is satisfied)}. \end{cases} \tag{8.43}$$

The exponent θ_{g} is a non-negative constant (typically, a value of $\theta_{\mathrm{g}} = 2$ is used) and ensures that, for $\theta_{\mathrm{g}} > 1$, the amount of penalty will increase at a faster rate than will the amount of violation of a constraint (*i.e.* the distance of the solution from feasibility, also called distance to feasibility (Smith & Coit, 1997)).

For **static penalties**, the penalty coefficients, $r_{h,k}$ and $r_{g,j}$, are considered as constant values and the penalty terms of the penalised function f_{p} are proportional to the value of the constraints violation ($g(\mathbf{x})$ and/or $h(\mathbf{x})$). A simpler variation of the static penalty method occurs for $\theta_{\mathrm{g}} = 0$, where the penalty terms become proportional to the number of violated constraints. Despite being simpler, this

last method presents the disadvantage of dealing with a discontinuous function, which difficult the optimisation problem (Rao, 2009).

The penalty coefficients may be difficult to determine, which can represent a drawback in the use of static penalties. The use of **dynamic penalties** allows to overcome such drawback, similarly to the sequential unconstrained minimisation technique (SUMT) used in gradient-based algorithms (Andrade-Campos et al., 2015). The idea of the dynamic penalty methods is to increase the penalty coefficient in each iteration (or generation, for instance) according to its proximity to the feasible solution. However, this method may lead to infeasible solutions if the penalty increasing rate is too slow or lead to non-optimal feasible solutions for a too fast increasing rates (Smith & Coit, 1997).

An approach to avoid the need of problem-specific tuning required in the two previous approaches is the use of **adaptive penalties** (Smith & Coit, 1997). One example of an adaptive penalty method is the use of a penalty function multiplier that is updated in generations intervals based on whether or not the best solution found was feasible during that interval (see more about adaptive approaches in Smith and Coit (1997)).

From the proposed optimisation problem previously described in Equation 8.12 and considering, for instance, an exterior penalty method with fixed penalty coefficients r_h and r_g, the augmented (penalised) objective function $C^\mathrm{p}_\mathrm{total}$ can be obtained by:

$$\begin{aligned} C^\mathrm{p}_\mathrm{total}(\mathbf{X}, r_\mathrm{h}, r_\mathrm{g}) = C_\mathrm{total}(\mathbf{X}) + r_\mathrm{h}\left[n_\mathrm{warn}(\mathbf{X})\right]^2 + r_\mathrm{g}\sum_{j=0}^{n_\mathrm{tanks}}\left[max\left\{0; L_{j,\mathrm{final}}(\mathbf{X}) - L_{j,\mathrm{initial}}(\mathbf{X})\right\}\right]^2 \\ + r_\mathrm{g}\sum_{j=0}^{n_\mathrm{tanks}}\left[max\left\{0; L_j(\mathbf{X}) - L_{j,\mathrm{max}}\right\}\right]^2 + r_\mathrm{g}\sum_{j=0}^{n_\mathrm{tanks}}\left[max\left\{0; L_{j,\mathrm{min}} - L_j(\mathbf{X})\right\}\right]^2 \\ + r_\mathrm{g}\sum_{k=0}^{n_\mathrm{cnodes}}\left[max\left\{0; P_{k,\mathrm{min}} - P_k(\mathbf{X})\right\}\right]^2 \end{aligned} \tag{8.44}$$

8.4 Comparing optimisation methods performance

An optimisation algorithm can be evaluated in terms of distinct (and sometimes contradictory) criteria, such as (Nocedal & Wright, 2006):

- Robustness, which presupposes the algorithm to perform well in a wide variety of problems, considering distinct initial solutions;
- Efficiency, related with the processing computer time or the storage requirements of the algorithms;
- Accuracy, which expects the algorithm to find a solution with precision and lower sensitivity to errors and/or approximations on data.

In order to be considered adequate for a certain problem, a specific algorithm should fulfil certain properties. For distinct optimisation problems, the importance of the previously defined criteria is variable. An analysis on the problem to be solved should be performed in order to understand which of the criteria is more relevant. This is important because, in some situations, those criteria may compete (*e.g.* an algorithm may be fast but loose in accuracy, which may not be desired for certain problems).

For the problems addressed in this work, it is important to use efficient and robust optimisation algorithms in order to obtain fast results (particularly in the case of real-time processes) and robust enough to perform well in a wide variety of networks configurations. Thus, in the results obtained for distinct problems, the computational time is compared and the feasibility of the obtained solutions is analysed.

9. Demand forecasting

The main forecasting techniques for time series data are described, including the traditional techniques based on regression analysis, exponential smoothing and time series analysis, as well as the innovative techniques based on artificial neural networks. Procedures to develop and evaluate a forecasting model are also presented.

In order to obtain feasible control optimisation results, it is of the most importance to use accurate data for predicting the behaviour of water distribution networks as close as possible to the reality. For an efficient operational control and management, the highly variable water demands need to be predicted in advance. For real-time operations, for instance, the process of predicting the near-future demands is critical, since the model of the network also need a real-time update.

Given the importance of this topic, the current chapter is devoted to the methodologies applied in this work for short-term water demand forecasting.

9.1 Forecasting techniques

Most forecasting problems involve the use of time series data (Montgomery, Jennings, & Kulahci, 2008), which corresponds to time-oriented observations on a specific variable (water demand, in this case).

Time series data are commonly described according to three main components: trend, season and cycle. The trend component is related to the long-term increase or decrease of data, while the seasonal component is related to patterns identified in the data that are affected by factors like the time of the year or the hour of the day (see examples of time series with and without trend and seasonality in Figure 9.1). Finally, the cycle is related to fluctuations in data without fixed periods. While seasonal patterns have a fixed and known length, cyclic patterns have variable and unknown length - seasonality with changing variation (Hyndman & Athanasopoulos, 2013).

While qualitative forecasting techniques are used in situations where there is little or no historical data (such as the example of forecasting the sales of a new product), quantitative forecasting techniques uses the information of historical data, predicting the future based on observed patterns and relationships in the data (Montgomery et al., 2008).

Concerning the quantitative techniques, Montgomery et al. (2008) enumerate and describe three main types of traditional forecasting techniques: (i) regression models, (ii) smoothing models and (iii) time series models. More innovative techniques based on artificial intelligence have been largely

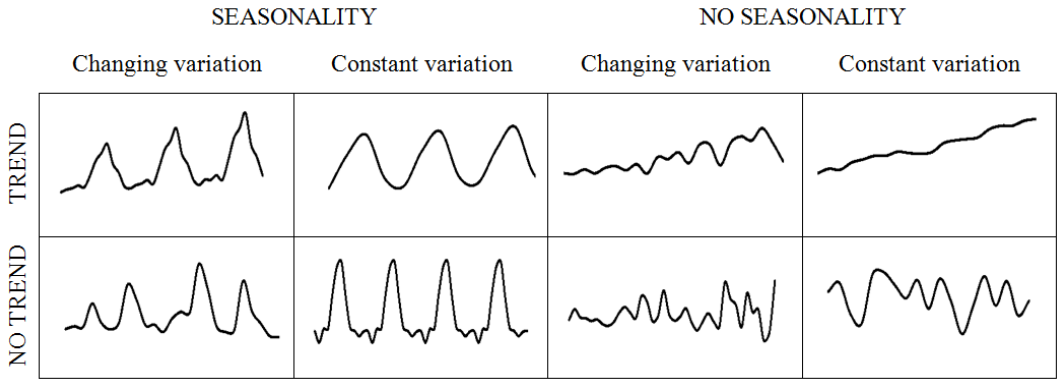


Figure 9.1: Examples of time series demonstrating seasonal and trend effects over the time.

applied in forecasting problems and compared to the traditional ones. It is the case of Artificial Neural Networks (ANN), Support Vector Regression (SVR), Fuzzy Logic (FL) and also hybrid methods combining distinct forecasting methods and/or incorporating optimisation algorithms.

9.1.1 Regression Analysis

Regression models are based in the relationship between the variable of interest (forecast variable, y) and other input variables (predictor variables, $\mathbf{z}$). This type of techniques may be simple and easy to implement but may also be limited in certain situations such as with nonlinear data and, specially, with very noisy data (Adamowski & Karapataki, 2010; Montgomery et al., 2008).

For time series data, a **Multiple Linear Regression** (MLR) model can be written as (Hyndman & Athanasopoulos, 2013; Montgomery et al., 2008):

$$y_t = \beta_0 + \beta_1 z_{1,t} + \beta_2 z_{2,t} + ... + \beta_k z_{k,t} + \varepsilon_t, \tag{9.1}$$

where k is the number of predictor variables, ε_t is the error between the O observed data points at time period t and the predicted ones defined by the regression line specified by the model. $\beta_0, \beta_1, ..., \beta_k$ are the unknown regression parameters (or model parameters) usually estimated using the Least Squares method (in linear models) by finding the values that minimise the Sum of Squared Errors (Hyndman & Athanasopoulos, 2013):

$$\sum_{t=1}^{O} \varepsilon_t^2 = \sum_{t=1}^{O} (y_t - \beta_0 - \beta_1 z_{1,t} - ... - \beta_k z_{k,t})^2. \tag{9.2}$$

Predictions of y can be calculated by ignoring the error in the regression equation (eq. 9.1) (Hyndman & Athanasopoulos, 2013):

$$\hat{y}_t = \hat{\beta}_0 + \hat{\beta}_1 z_{1,t} + \hat{\beta}_2 z_{2,t} + ... + \hat{\beta}_k z_{k,t}, \tag{9.3}$$

where $\hat{\beta}_0, \hat{\beta}_1, ..., \hat{\beta}_k$ are the already estimated regression coefficients according to equation 9.2.

When these predictions are made using the same predictor data that were used to estimate the model, $\hat{y}_t$ correspond to fitted data. If different predictor data are used (validation data set), then $\hat{y}_t$ is considered a real forecast.

The residuals of these predictions (or forecasting errors) are given by the difference between the observed and the predicted data ($\varepsilon_t = y_t - \hat{y}_t$).

It is important to notice that equation 9.1 expresses only the relationship between a single value for the predictor and forecast variables. In order to represent all the values, the model should be written in a matrix form, and the MLR model would become (Montgomery et al., 2008):

$$\mathbf{y} = \boldsymbol{\beta}\mathbf{Z} + \boldsymbol{\varepsilon}, \tag{9.4}$$

where $\mathbf{y}$ is a $(O \times 1)$ vector of the observations, $\mathbf{Z}$ is a $(O \times k)$ matrix of the predictor variables, $\boldsymbol{\beta}$ is a $(k \times 1)$ vector of the regression coefficients and $\boldsymbol{\varepsilon}$ is a $(O \times 1)$ vector of random errors. Therefore, in the matrix notation, the forecasts can be determined by:

$$\hat{\mathbf{y}} = \hat{\boldsymbol{\beta}}\mathbf{Z}, \tag{9.5}$$

where $\hat{\boldsymbol{\beta}} = (\mathbf{Z}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{Z}^{\mathrm{T}}\mathbf{y}$ can be estimated by least-square techniques (Montgomery et al., 2008).

For simplification, the index notation will be used in this work instead of the matrix notation.

In the **Simple Linear Regression** model, only one predictor variable z is considered and the model can be simply written as (Hyndman & Athanasopoulos, 2013):

$$y_t = \beta_0 + \beta_1 z_t + \varepsilon_t, \tag{9.6}$$

where β_0 and β_1 represent, respectively, the interception and the slope coefficients of the regression line that defines the linear relationship between data.

In some situations, the use of linear relationships may not be adequate. In these cases, the predictor variables and/or the forecast variables can be replaced by a transformation of such variables. A **Multiple Non-Linear Regression** (MNLR) model can then be represented by:

$$f^{\mathrm{NL}}(y_t) = \beta_0 + \beta_1 f_1^{\mathrm{NL}}(z_{1,t}) + \beta_2 f_2^{\mathrm{NL}}(z_{2,t}) + ... + \beta_k f_k^{\mathrm{NL}}(z_{k,t}) + \varepsilon_t, \tag{9.7}$$

where both $f^{\mathrm{NL}}(y)$ and $f_1^{\mathrm{NL}}(z_1), ..., f_k^{\mathrm{NL}}(z_k)$ are possibly non-linear functions (example: $f^{\mathrm{NL}}(z) = \log z$).

9.1.2 Exponential Smoothing Methods

Smoothing models use a function obtained from previous observations to predict future ones (Montgomery et al., 2008). This technique of obtaining a smooth function (exponential smoother) from the data can be attractive to deal with noisy data.

The simplest method is called **Simple Exponential Smoothing** (SES, or first-order exponential smoothing) and is suitable for forecasting data with no trend and no seasonality. With this method, forecasts are calculated using weighted averages where the weights decrease exponentially as observations move away from the present, *i.e.* smaller weights are associated to the oldest observations (Hyndman & Athanasopoulos, 2013). To modelling the series with SES, the first-order exponential smoother is used and the model can be written as (Montgomery et al., 2008):

$$y_t = L_t^{\mathrm{s}} + \varepsilon_t = \alpha^{\mathrm{s}} y_t + (1 - \alpha^{\mathrm{s}}) L_{t-1}^{\mathrm{s}} + \varepsilon_t \tag{9.8}$$

where L_t^s, also known as level component (Hyndman & Athanasopoulos, 2013), represents the smoothed value of the observation at current time t, $0 \leq \alpha^s \leq 1$ is the discount factor (or smoothing parameter) and $(1-\alpha^s)$ represents the weight of the smoothed value of the previous observations.

In equation 9.8, the initial smoothed value L_0^s and the value of α^s need to be estimated. According to Hyndman and Athanasopoulos (2013), the two commonly used estimates for L_0^s are (i) the first value of the observed series, $L_0^s = y_1$ (when early and fast changes are expected to occur in the process), or (ii) the average of the observed data, $L_0^s = \bar{y}$ (when the process is locally constant at least at the beginning). The value of α^s is commonly determined by finding the value that minimises the Sum of the Squared Errors (SSE) of the fitted model:

$$\sum_{t=1}^{O} \varepsilon_t^2 = \sum_{t=1}^{O} (y_t - L_t^s)^2. \tag{9.9}$$

When $\alpha^s = 1$, the model is not smoothed and equal to the observed series, *i.e.* $L_t^s = y_t$. Smaller values of α^s leads to more smoothed functions.

Using the SES model, the forecast at time $t+h$ (h-step ahead) is equal to the current value of the exponential smoother (Montgomery et al., 2008):

$$\hat{y}_{t+h} = L_t^s = \alpha^s y_t + (1-\alpha^s) L_{t-1}^s. \tag{9.10}$$

For **higher order exponential smoothing** methods the procedure is similar. An σ-order exponential smoother can be written as (Montgomery et al., 2008):

$$L_t^{s(\sigma)} = \alpha^s L_t^{s(\sigma-1)} + (1-\alpha^s) {L_{t-1}^s}^{(\sigma)}, \tag{9.11}$$

where the initial value $L_0^{s(\sigma)} = L_1^{s(\sigma-1)}$.

The problem with these simple forecasting methods is that they are not suitable for data that exhibit cyclical or seasonal patterns. However, Holt (in 1957) and Winters (in 1960) proposed two distinct approaches based on adjustments to these methods in order to deal with seasonal data, namely, an additive seasonal model and a multiplicative seasonal model (Hyndman & Athanasopoulos, 2013; Montgomery et al., 2008).

The Holt-Winters Seasonal models use three smoother functions that represents three components of a time series: (i) the level component, L_t^s, (ii) the trend component, T_t^s, and (iii) the seasonal component, S_t^s. The difference between the two proposed models is related with the nature of the seasonal component. While the additive seasonal model is preferred when seasonal variations are roughly constant through the series, the multiplicative seasonal model works better when the seasonal variations change proportionally to the level of the series (Hyndman & Athanasopoulos, 2013).

The equation for modelling the series using the **Holt-Winters Additive Seasonal model** is:

$$y_t = L_t^s + T_t^s + S_t^s + \varepsilon_t. \tag{9.12}$$

The level, trend and seasonal smoothers can be respectively written as (Hyndman & Athanasopoulos, 2013):

$$L_t^s = \alpha_1^s (y_t - L_{t-m}^s) + (1-\alpha_1^s)(L_{t-1}^s + T_{t-1}^s), \tag{9.13}$$

$$T_t^s = \alpha_2^s(L_t^s - L_{t-1}^s) + (1-\alpha_2^s)T_{t-1}^s \quad \text{and} \tag{9.14}$$

$$S_t^s = \alpha_3^s(y_t - L_{t-1}^s) + (1-\alpha_3^s)S_{t-m}^s, \tag{9.15}$$

where m represents the period of seasonality and α_1^s, α_2^s and α_3^s are smoother parameters with ranges between 0 and 1. The estimation of these parameters can also be done by finding the values that minimise the SSE of the fitted model:

$$\sum_{t=1}^{O} \varepsilon_t^2 = \sum_{t=1}^{O} (y_t - L_t^s - T_t^s - S_t^s)^2. \tag{9.16}$$

The initial values for each smoother, L_0^s, T_0^s and S_0^s, can be estimated by (Hyndman & Athanasopoulos, 2013):

$$L_0^s = \frac{1}{m}(y_1 + ... + y_m), \tag{9.17}$$

$$T_0^s = \frac{1}{m}\left[\frac{y_{m+1} - y_1}{m} + ... + \frac{y_{m+m} - y_m}{m}\right] \quad \text{and} \tag{9.18}$$

$$S_0^s = y_m - L_o^s, \quad S_{-1}^s = y_{m-1} - L_0^s, \quad ..., \quad S_{-m+1}^s = y_1 - L_0^s. \tag{9.19}$$

The forecast equation to predict an h-step ahead using the Holt-Winters Additive Seasonal model is (Hyndman & Athanasopoulos, 2013):

$$\hat{y}_{t+h} = L_t^s + hT_t^s + S_{t-m+h}^s \tag{9.20}$$

and the forecast error in each step is calculated by:

$$\varepsilon_t = y_t - \hat{y}_t = y_t - L_{t-1}^s - T_{t-1}^s - S_{t-m}^s. \tag{9.21}$$

Concerning the **Holt-Winters Multiplicative Seasonal model**, the equation for modelling the series is:

$$y_t = (L_t^s + T_t^s)S_t^s + \varepsilon_t. \tag{9.22}$$

In this case, the level, trend and seasonal smoothers are respectively written as (Hyndman & Athanasopoulos, 2013):

$$L_t^s = \alpha_1^s\left(\frac{y_t}{S_{t-m}^s}\right) + (1-\alpha_1^s)(L_{t-1}^s + T_{t-1}^s), \tag{9.23}$$

$$T_t^s = \alpha_2^s(L_t^s - L_{t-1}^s) + (1-\alpha_2^s)T_{t-1}^s \quad \text{and} \tag{9.24}$$

$$S_t^s = \alpha_3^s\left(\frac{y_t}{L_{t-1}^s}\right) + (1-\alpha_3^s)S_{t-m}^s, \tag{9.25}$$

and the initial values for each smoother, L_0^s, T_0^s and S_0^s, are estimated by (Hyndman & Athanasopoulos,

2013):

$$L_0^s = \frac{1}{m}(y_1 + ... + y_m), \tag{9.26}$$

$$T_0^s = \frac{1}{m}\left[\frac{y_{m+1} - y_1}{m} + ... + \frac{y_{m+m} - y_m}{m}\right] \quad \text{and} \tag{9.27}$$

$$S_0^s = \frac{y_m}{l_o}, \quad S_{-1}^s = \frac{y_{m-1}}{L_0^s}, \quad ..., \quad S_{-m+1}^s = \frac{y_1}{L_0^s}. \tag{9.28}$$

The forecast equation to predict an h-step ahead using the Holt-Winters Multiplicative Seasonal model is (Hyndman & Athanasopoulos, 2013):

$$\hat{y}_{t+h} = (L_t^s + hT_t^s)S_{t-m+h}^s \tag{9.29}$$

and the forecast error in each step is calculated by:

$$\varepsilon_t = y_t - \hat{y}_t = y_t - (L_{t-1}^s - T_{t-1}^s)S_{t-m}^s. \tag{9.30}$$

9.1.3 Time Series Analysis

Forecasting models based on time series analysis consist in the analysis of statistical properties of stationary (constant mean and variance) and stochastic processes/models using the historical data.

As mentioned by Montgomery et al. (2008), the methods based on exponential smoothing may be inefficient and sometimes inappropriate for not taking advantage of the serial dependence in the observations by the most effective way. The Auto-Regressive Integrated Moving Average (ARIMA) models provide another approach to time series forecasting. While exponential smoothing models are based on the description of trend and seasonality of data, ARIMA models are based on the description of autocorrelation in data (Hyndman & Athanasopoulos, 2013).

Autoregressive (AR) methods can be compared to the previously explained multiple linear regression (MLR). While in MLR the forecast is based on a linear relationship between the variable of interest and the predictor variables, AR methods consider a relationship with the past values (lags) of the variable of interest instead of the predictor variables (Hyndman & Athanasopoulos, 2013).

An autoregressive model of order ϕ, AR(ϕ), where ϕ represents the number of lags, can be written as (Hyndman & Athanasopoulos, 2013; Montgomery et al., 2008):

$$y_t = \mu_0 + \mu_1 y_{t-1} + \mu_2 y_{t-2} + ... + \mu_\phi y_{t-\phi} + \varepsilon_t, \tag{9.31}$$

where ν_0 is a constant that represents the series level (similar to the intercept parameter of a regression line), $\mu_1, \mu_2, ..., \mu_\phi$ are regression parameters and ε_t is an error term (usually called white noise).

Moving Average (MA) methods also follow a similar approach to the multiple regression but uses past forecast errors instead of predictor variables. A model based on this type of method with order ψ (ψ lags) can be written as (Hyndman & Athanasopoulos, 2013; Montgomery et al., 2008):

$$y_t = \theta_0 + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + ... + \theta_\psi \varepsilon_{t-\psi} + \varepsilon_t, \tag{9.32}$$

where θ_0 represents a constant (level) and $\theta_1, \theta_2, ..., \theta_\psi$ represent regression parameters.

The general model of a combined **Autoregressive Moving Average** (ARMA), usually referred as ARMA(ϕ, ψ), where ϕ is the order of the autoregressive part and ψ the order of the moving average part, is given as (Montgomery et al., 2008):

$$\begin{aligned} y_t &= \delta_0 + \mu_1 y_{t-1} + \mu_2 y_{t-2} + ... + \mu_\phi y_{t-\phi} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + ... + \theta_\psi \varepsilon_{t-\psi} \\ &= \delta + \sum_{i=1}^{\phi} \mu_i y_{t-i} + \varepsilon_t + \sum_{i=1}^{\psi} \theta_i \varepsilon_{t-i}. \end{aligned} \tag{9.33}$$

Before applying any of the previously mentioned models (AR, MA or ARMA), three steps must be performed: (i) the original time series y_t must be transformed to become stationary around its mean and variance (time series with trends or seasonality are not stationary since these components affect the value of the series at different times), (ii) the appropriate order of both ϕ and ψ must be specified and (iii) the value of the regression parameters $\mu_1, ..., \mu_\phi$ and $\theta_1, ..., \theta_\psi$ must be estimated (Makridakis & Hibon, 1997).

Generally, the differencing method is used to achieve stationarity in the time series mean (Hyndman & Athanasopoulos, 2013; Makridakis & Hibon, 1997; Montgomery et al., 2008). This method consists in the computation of the differences between consecutive observations, allowing the stabilisation of the sample mean. A first order differencing is computed by:

$$y'_t = y_t - y_{t-1}, \tag{9.34}$$

while the second order differencing will be dependent on the first order differences:

$$\begin{aligned} y''_t &= y'_t - y'_{t-1}, \\ &= (y_t - y_{t-1}) - (y_{t-1} - y_{t-2}). \end{aligned} \tag{9.35}$$

The same approach is followed for consecutive differencing calculations.

When dealing with seasonal data, it is preferable to compute seasonal differencing, which is given by the difference between an observation and the corresponding observation from the previous season (Hyndman & Athanasopoulos, 2013):

$$y'_t = y_t - y_{t-m}, \tag{9.36}$$

where m represents the number of periods of the season.

In order to stabilise the variance of the time series, the use of variables transformations is common (mostly logarithmic and power transformations). According to Montgomery et al. (2008), a popular type of data transformation is the power family, given by:

$$y^{(\lambda_P)} = \begin{cases} \frac{y^{\lambda_P} - 1}{\lambda_P \dot{y}^{\lambda_P - 1}}, & \lambda_P \neq 0 \\ K_y \ln y, & \lambda_P = 0 \end{cases}, \tag{9.37}$$

where $K_y = \exp\left[(\frac{1}{O}) \sum_{t=1}^{O} \ln y_t\right]$ is the geometric mean of the observations. This equation allows distinct transformations of data according to the defined power parameter λ_P. When no transformation is intended, λ_P should be set to 1. The typical transformations are: square root ($\lambda_P = 0.5$), logarithmic

($\lambda_P = 0$), reciprocal square root ($\lambda_P = -0.5$) and inverse ($\lambda_P = -1$).

The autoregressive and moving average orders (ϕ and ψ, respectively) can be specified by plotting the autocorrelation and partial autocorrelation functions (ACF and PACF) of the series. Autocorrelation is a linear dependence of a variable with itself at two points in time. Thus, the autocorrelation of series gives the correlation between y_t and y_{t-h}, where h represents the lag. For a time series $y_1, y_2, ..., y_O$, the autocorrelation coefficient at lag h is given by (*MathWorks: Autocorrelation and Partial Autocorrelation*, 2015; Montgomery et al., 2008):

$$\rho_h = \frac{\text{Cov}(y_t, y_{t-h})}{\text{Var}(y_t)} = \frac{\frac{1}{O}\sum_{t=h+1}^{O}(y_t - \bar{y})(y_{t-h} - \bar{y})}{\frac{1}{O}\sum_{t=1}^{O}(y_t - \bar{y})^2}, \tag{9.38}$$

where $\text{Cov}(y_t, y_{t-h})$ is the auto-covariance at lag h and $\text{Var}(y_t)$ is the sample variance. The collection of values of ρ_h, with $h = 0, 1, 2, ..., O-1$ is the autocorrelation function (ACF).

The correlation between two variables can result from a mutual linear dependence on other variables. However, the partial autocorrelation provides the correlation between two variables after removing any linear dependence on other variables.

For a time series, the partial autocorrelation between y_t and y_{t-h} (or the partial lag-h autocorrelation) is defined as the conditional correlation between y_t and y_{t-h}, conditional on $y_{t-h+1}, ..., y_{t-1}$, the set of observations that come between the time points t and $t-h$ (Montgomery et al., 2008; *STAT 510 – Applied Time Series Analysis*, 2015):

$$\varphi_h = \frac{\text{Cov}(y_t, y_{t-h} | y_{t-h+1}, ..., y_{t-1})}{\text{Var}(y_t | y_{t-h+1}, ..., y_{t-1})}. \tag{9.39}$$

The PACF is the sequence φ_h, with $h = 0, 1, 2, ..., O-1$.

Both denominators of equations 9.38 and 9.39 are approximations, considering that the series are stationary.

The ACF is commonly used to identify the ψ order of a MA process since it is expected to "cut off" after lag ψ, which means that the ACF should be zero for $h > \psi$. In turn, the PACF is employed in AR processes since it is expected to "cut off" after lag ϕ (Montgomery et al., 2008).

Finally, the regression parameters are usually determined using the optimisation procedure that minimises the sum of square errors or other appropriate error function.

Autoregressive Integrated Moving Average (ARIMA) models, also known as Box-Jenkins models, represent a combination of differencing, moving average and auto-regression. The term "Integrated" is related to the reverse of differencing. An ARIMA(ϕ, υ, ψ) model can be represented as (Hyndman & Athanasopoulos, 2013):

$$y'_t = \delta + \mu_1 y'_{t-1} + \mu_2 y'_{t-2} + ... + \mu_\phi y'_{t-\phi} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + ... + \theta_\psi \varepsilon_{t-\psi}, \tag{9.40}$$

where y'_t represents the differenced series, which may have been differenced more than once depending on the differencing order υ.

To develop an ARIMA model, the main steps should be: (i) stabilise the variance using data transformation, (ii) apply differencing until the series appear stationary, (iii) plot the ACF and PACF in order to determine the model orders (ϕ and ψ) and, finally, (iv) determine the model coefficients (regression parameters).

In order to forecast using an ARIMA model, equation 9.40 should be expanded and re-written in order to obtain y_t in the left side and the other terms in the right. Thus, an ARIMA h-step ahead forecast can be written as (Hyndman & Athanasopoulos, 2013; Montgomery et al., 2008):

$$\hat{y}_{t+h} = \delta + \sum_{i=1}^{\phi+\upsilon} \mu_i y_{t+h-i} + \varepsilon_{t+h} + \sum_{i=1}^{\psi} \theta_i \varepsilon_{t+h-i}. \tag{9.41}$$

ARIMA models can also be used for seasonal data by including seasonal terms in the previously mentioned model. This is commonly called **Seasonal Autoregressive Integrated Moving Average** and it is usually represented as ARIMA $(\phi,\upsilon,\psi)(\Phi,\Upsilon,\Psi)_m$, where $(\Phi,\Upsilon,\Psi)_m$ represents the seasonal part of the model and m is the number of periods per season.

Using the back-shift notation (also called lag operator notation), where $B_{\mathrm{s}}^i y_t = y_{t-i}$, an ARMA$(\phi,\psi)$ general model (equation 9.33) can be written in the following form (Hyndman & Athanasopoulos, 2013; *MathWorks: arima class*, 2015):

$$\beta(B_{\mathrm{s}}) y_t = \delta + \theta(B_{\mathrm{s}})\varepsilon_t, \tag{9.42}$$

where

$$\beta(B_{\mathrm{s}}) = 1 - \mu_1 L - \mu_2^2 L^2 - \ldots - \mu_\phi^\phi L^\phi, \tag{9.43}$$

and

$$\theta(B_{\mathrm{s}}) = 1 + \theta_1 L + \theta_2^2 L^2 + \ldots + \theta_\psi^\psi L^\psi. \tag{9.44}$$

Following the same notation, a general ARIMA (ϕ,υ,ψ) can be written as (Hyndman & Athanasopoulos, 2013):

$$\beta(B_{\mathrm{s}})(1-B_{\mathrm{s}})^\upsilon y_t = \delta + \theta(B_{\mathrm{s}})\varepsilon_t, \tag{9.45}$$

where $(1-B_{\mathrm{s}})^\upsilon$ is the υ^{th}-order difference.

Finally, a Seasonal ARIMA $(\phi,\upsilon,\psi)(\Phi,\Upsilon,\Psi)_m$, including differencing, multiplicative seasonality and seasonal differencing, can be obtained by multiplying the seasonal terms (*MathWorks: arima class*, 2015; Montgomery et al., 2008):

$$\mu^*(B_{\mathrm{s}}^m)(1-B_{\mathrm{s}}^m)^\Upsilon \mu(B_{\mathrm{s}})(1-B_{\mathrm{s}})^\upsilon y_t = \delta + \theta^*(B_{\mathrm{s}}^m)\theta(B_{\mathrm{s}})\varepsilon_t, \tag{9.46}$$

where

$$\mu^*(B_{\mathrm{s}}^m) = 1 - \mu_1^* L^m - \mu_2^{*2} L^{2m} - \ldots - \mu_\Phi^{*\Phi} L^{\Phi m}, \tag{9.47}$$

and

$$\theta^*(B_{\mathrm{s}}^m) = 1 + \theta_1^* L^m + \theta_2^{*2} L^{2m} + \ldots + \theta_\Psi^{*\Psi} L^{\Psi m}. \tag{9.48}$$

9.1.4 Artificial Intelligence

Traditional statistical methods can be limited with non-linear relationships and very noisy data. For this reason, models based on artificial intelligence, capable of identifying complex and non-linear phenomena/behaviours, have been largely applied.

In artificial intelligence, the study of pattern recognition and computational learning theory is called machine learning (or computational intelligence). The most used machine learning approaches

for time series forecast are typically the Artificial Neural Networks (ANN). However, the use of Support Vector Machines (SVM) and Fuzzy Logic approaches or even hybrid approaches has recently become popular. Applied to time series forecasting, these types of techniques basically operates by processing historical data (or other type of input data) and building a data-driven model capable of solve prediction problems. Such data-driven models are trained on a set of input and target output describing the phenomena in question (Solomatine & Siek, 2006).

Artificial Neural Networks (ANN) are based on mathematical models inspired in the way the human brain process information. As described by Robert Hecht-Nielsen, a pioneer in artificial intelligence and neural networks, a neural network is (Caudill (1987) *apud A Basic Introduction To Neural Networks* (2015)):

> "*...a computing system made up of a number of simple, highly interconnected processing elements, which process information by their dynamic state response to external inputs.*"

With around 10 billion neurons (or elements) in the cortex and 60 trillion connections between them, the human brain presents a very complex, non-linear and parallel structure which makes it very efficient for information processing, learning and reasoning (Montgomery et al., 2008).

An ANN model consists of two or more layers: (i) an input layer, (ii) an output layer and, optionally, (iii) one or more intermediary layers called hidden layers. Each layer consists of multiple nodes (also called neurons or elements) that represents the variables of the model (see Figure 9.2).

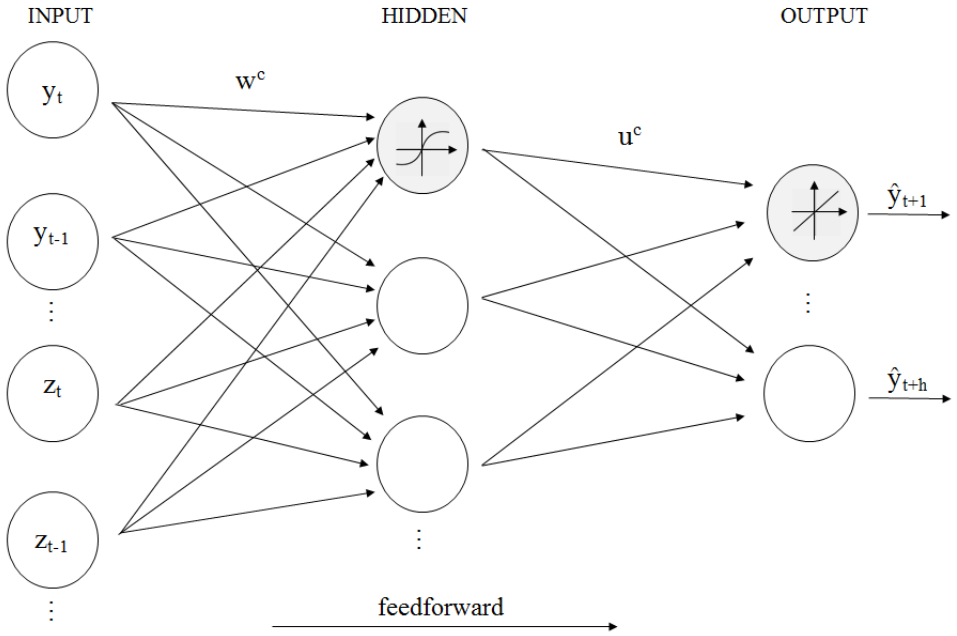


Figure 9.2: Scheme representing an example of a 2-layer feed-forward artificial neural network for time series forecasting. The input layer may contain the lags of the variable to predict $(y_t, y_{t-1}, ...)$ as well as other predictors $(z_t, z_{t-1}, ...)$ and the output can have a single or multiple neurons according to the defined time horizon (1 to h steps ahead).

Each node of the network receives information from the previous layer* (the input layer or an hidden layer) as a linear combination of each node output, according to the connection weights u^c

*This is the called feed-forward network. Feed-back connections can also be used. Feed-back neural networks, also

and *w*c (parameters that must be estimated) defined in each connection, and then returns an output that is represented by a transformation of such combined information through an activation function (also called transfer function). This output can be used for the next layers (and the feed-forward process is repeated) or be the expected model output.

Each node output $\hat{y}_k$ of a 2-layer ANN model (2 layers of connections) with n_{in} input nodes, n_{hidden} hidden nodes and n_{out} output nodes can be represented as:

$$\hat{y}_k = f_1^{\text{A}} \left(\sum_{j=1}^{n_{\text{hidden}}} u_{j,k}^{\text{c}} f_2^{\text{A}} \left(\sum_{i=1}^{n_{\text{in}}} w_{i,j}^{\text{c}} z_i + \theta_j^{\text{b}} \right) + \theta_k^{\text{b}} \right), \tag{9.49}$$

where $i = 1, ..., n_{\text{in}}$, $j = 1, ..., n_{\text{hidden}}$ and $k = 1, ..., n_{\text{out}}$. z_i and $\hat{y}_k$ represents, respectively, the i^{th} model input and the k^{th} model output, θ^{b} is a parameter that represents an intercept in linear regression (usually called the bias node) and f_1^{A} and f_2^{A} are activation functions. The activation functions are usually sigmoidal (S shaped) or linear (Montgomery et al., 2008). Considering, for example, f_1^{A} as a log-sigmoid function, $f_1^{\text{A}}(z) = \frac{1}{1+e^{-z}}$, and f_2^{A} as a linear function, $f_2^{\text{A}}(z) = z$, then equation 9.49 would take the following form:

$$\hat{y}_k = \sum_{j=1}^{n_{\text{hidden}}} \left(u_{j,k}^{\text{c}} \frac{1}{1 + exp\left(\sum_{i=1}^{n_{\text{in}}} w_{i,j}^{\text{c}} x_i + \theta_j^{\text{b}} \right)} \right) + \theta_k^{\text{b}}, \tag{9.50}$$

The use of non-linear activation functions (such as sigmoid or hyperbolic tangent functions) in the hidden layers is commonly preferable since they tend to reduce the effect of extreme input values, thus making the network somewhat robust to outliers (Hyndman & Athanasopoulos, 2013). Recently, Radial Basis functions (RBF), where $f^{\text{A}}(z) = e^{-z^2}$, have also been used.

Note that the simplest ANN model, a single-layer neural network, with no hidden layers, is equivalent to a linear regression model (Hyndman & Athanasopoulos, 2013), where the predictor variables (in the input layer) are combined through weights (like the regression parameters) to compute the output (forecast).

In order to estimate the model parameters (weights and bias) that fit the data, a set of inputs and target outputs should be initially provided to the model (supervised learning[†]). Thus, the training/learning process (parameter estimation) begins typically by minimising the overall residual sum of squares (or the mean squared errors) taken over all responses (target outputs) and observations (inputs), which is a non-linear Least Squares problem (Hyndman & Athanasopoulos, 2013; Montgomery et al., 2008).

A popular learning method is the called Back-Propagation algorithm, which looks for the minimum of the error function in weight space using gradient-based optimisation methods. The name Back-Propagation is related to the backward propagation of errors through the layers of the network (Atiya, 1991; Rojas, 1996). Although the steepest descent algorithm is typically associated to the Back-propagation method, other derivative-based optimisation algorithms, such as the Levenberg-Marquardt (LM) or the Conjugate Gradient (CG), can also be employed to find the minimum of the

called recurrent neural networks, allow the information to be sent from one layer to the previous ones - connections in opposite directions.

[†]Note that it is possible to train a neural network using unsupervised learning methods, *i.e.* with unknown target outputs. However, such methods will not be covered in this work (see more about this methods in Rojas, 1996).

error function.

The initial values for the model parameters are commonly defined randomly and then are updated/adjusted through the iterative learning process using the observed data. In ANNs, each iteration of weights update is usually called epoch. It is common to set a maximum number of epochs to stop the training process in case of non-convergence.

For the choice of the most adequate network architecture (number of layers, number of nodes and activation functions form), *trial and error* procedures or optimisation methods can be used.

9.2 Developing a forecasting model

The development of a forecasting model involves several steps that goes from the knowledge of the problem to the implementation of the developed model. The main stages of the process are briefly described.

1. Problem definition

   It is important to understand the problem, to know who requires the forecasts, how the forecasted data will be used and, finally, define some parameters such as the forecast horizon, forecast interval and level of accuracy required since all these will influence the model selection.

2. Data collection and selection

   Two types of information are useful and should be collected: (i) historical statistical data and (ii) information obtained from the expertise of who collect the data, which can provide essential explanations for some atypical historical data, helping in the data analysis. Not all historical data is useful for a specific problem. Data should be selected according to the problem and the method to be followed.

3. Data analysis and pre-processing

   This is a crucial step for the decision of the forecasting model. At a first stage, it is important to plot the time series for the recognition of missing data, potential outliers, trends and/or seasonal patterns. After that, measures to deal with missing data, outliers and other data-related problems should be applied. Meanwhile, it may be useful to obtain numerical summaries of the data such as the sample mean, standard deviation, percentiles, correlations, etc.

   In order to identify potential predictor variables (such as weather or anthropic variables), scatter plots can be very valuable.

4. Model selection, fitting/training and validation

   According to the available data and the previous mentioned analysis and pre-processing of such data, one or more forecasting models should be selected. After that, the data is commonly split into two or three sets: (i) fitting/training set and validation set or, alternatively (and recommended), (ii) fitting/training set, cross-validation set and validation set. The cross-validation data set can be used to monitor the model performance during the fitting process. When using, for instance, forecasting methods based on ANN, the cross-validation plays a crucial role.

Since during training weights are being activated and adjusted in order to minimise the error, at a certain point, the network activates more weights than necessary and starts to fit the noise of the training data. Monitoring this data with a distinct data set allows to identify the best period to stop training the networks (see Figure 9.3) and to avoid over-fitting.

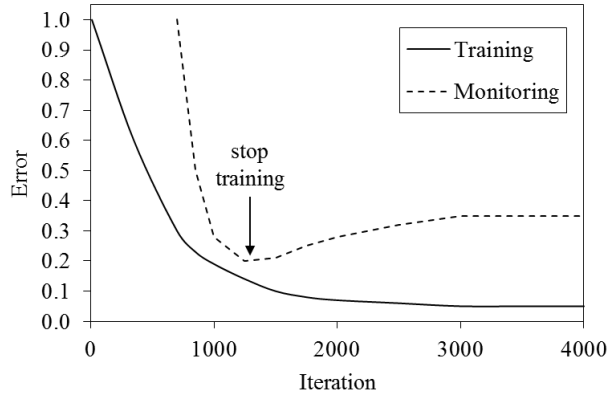


Figure 9.3: Scheme demonstrating the evolution of the training error and the error during monitoring using a distinct data set (cross-validation) [Adapted from (Luk et al., 2000)].

5. Model deployment and evaluation

   After the model selection and parameters estimation, it can finally be used to make forecasts. However, after implementation, a monitoring procedure should be performed since the models performance tends to deteriorate over time. For that, after the data for the forecast period become available, the forecast errors should be computed. Montgomery et al. (2008) propose the use of control charts of the forecast errors.

9.3 Evaluating forecasting models performance

The performance of a forecasting model can be defined (i) according to how well the model fits the sample data (in training/fitting process) or (ii) according to the capability of the forecasting technique to predict future observations (in test or validation processes)(Montgomery et al., 2008). In both cases, lower residuals and lower errors correspond to higher model performance (better fit and higher accuracy). Forecast accuracy is evaluated not only to validate the model but also to compare distinct models for selection (Montgomery et al., 2008).

According to Hyndman and Athanasopoulos (2013), good forecasting methods should present uncorrelated and zero mean residuals (difference between the observed/measured and the fitted data). If the residuals have a mean other than zero (biased forecasts) and/or the residuals are correlated, this suggests that it may contain information that is not being used and the method can be improved. However, this does not mean that a method that already fulfils these conditions cannot be improved.

The performance measure mostly used for the models fit evaluation is the Coefficient of Determination, more known as the Nash-Sutcliffe Model Efficiency (NSE), which is given by (Bennett et al., 2013):

$$NSE = 1 - \frac{\sum_{t=1}^{O} (y_t - \hat{y}_t)^2}{\sum_{t=1}^{O} (y_t - \bar{y})^2}, \tag{9.51}$$

where y_t is the t^{th} measured (or observed) value of the time series, $\hat{y}_t$ is the forecasted/fitted value for time t, $\bar{y} = \frac{1}{O}\sum_{t=1}^{O} y_t$ is the arithmetic mean of the measured values and O is the number of observations. $(y_t - \hat{y}_t)$ is the forecast error and $(y_t - \bar{y})$ is the standard deviation of the observed values. This metric compares the model performance to a model that only uses the mean of the observed data. The range is $(-\infty, 1)$ with 1 indicating the best performance. Negative values suggest that the model is worse than the one based on the mean.

The NSE metric may be confused with the Coefficient of Determination[‡] (R^2) that is given by the square of the Pearson Product Moment Correlation (PPMC) (Bennett et al., 2013):

$$R^2 = (PPMC)^2 = \left(\frac{\sum_{t=1}^{O}(y_t - \bar{y})(\hat{y}_t - \bar{y})}{\sqrt{\sum_{t=1}^{O}(y_t - \bar{y})^2}\sqrt{\sum_{t=1}^{O}(\hat{y}_t - \bar{y})^2}} \right)^2. \tag{9.52}$$

These two metrics measure the correlation of the measured and the fitted values, where the value 1 corresponds to a perfect correlation. *PPMC* ranges from -1 to 1. For being based on correlations between variables, these metrics are not indicated for the performance evaluation of forecasting models. As can be seen in figure 9.4, a good correlation does not mean that the model is good. In this case, the forecasted and the measured values are perfectly correlated ($R^2 = 1$) but the forecasted values are always inferior to the measured ones at the same time. In turn, the efficiency performance measure *NSE* demonstrates the poor performance of the forecasting model ($NSE = -2.8$).

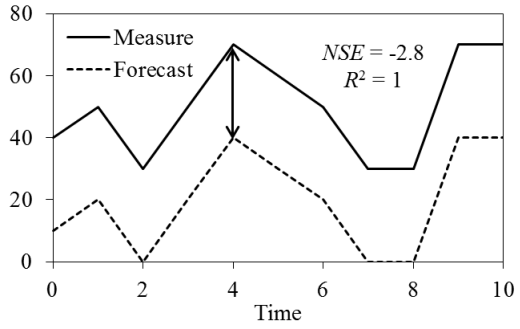


Figure 9.4: Demonstration of a forecasting model that presents a perfect correlation with measured variables ($R^2 = 1$) but does not represents a good forecasting model ($NSE = -2.8$) [Adapted from (Armstrong, 2001)].

Hyndman and Athanasopoulos (2013) describe others forecasting accuracy measures, grouping them into three main types based on: (i) scale-dependent errors, (ii) percentage errors and (iii) scaled errors. Due to its simplicity, the scale-dependent errors are good for comparing distinct forecast methods on a single data set. When comparing between distinct data sets, the percentage errors are preferable, however, this errors can not be computed if the data set contain zeros. In such cases, the alternative is the use of scaled errors.

The two most used scale-dependent measures are the Mean Absolute Error:

$$MAE = \frac{1}{O}\sum_{t=1}^{O} |y_t - \hat{y}_t|, \tag{9.53}$$

[‡]In fact, in regression analysis, NSE can be approximated to R^2 assuming that the sample covariance between $\hat{y}_i$ and $y_i - \hat{y}_i$ is zero.

and the Root Mean Square Error:

$$RMSE = \sqrt{\frac{\sum_{t=1}^{O}(y_t - \hat{y}_t)^2}{O}}. \tag{9.54}$$

Both metrics are in the same units of the sample data, which facilitates the interpretation. Although not mentioned by Hyndman and Athanasopoulos (2013), another measure within this group that may be helpful for the model evaluation is the Absolute Maximum Error (Bennett et al., 2013):

$$AME = \max|y_t - \hat{y}_t|. \tag{9.55}$$

The performance measure based on normalised errors most commonly used is the Mean Absolute Percent Error (also known as Mean Absolute Relative Error, MARE) (Hyndman & Athanasopoulos, 2013):

$$MAPE = \frac{1}{O}\sum_{t=1}^{O}\left|\frac{y_t - \hat{y}_t}{y_t}\right| \times 100\ \%. \tag{9.56}$$

Concerning the performance measures based on scaled error, Hyndman and Athanasopoulos (2013) propose the use of the Mean Absolute Scaled Error:

$$MASE = \frac{1}{O}\sum_{t=1}^{O}\left|\frac{y_t - \hat{y}_t}{\frac{1}{O_t - m}\sum_{i=m+1}^{O_t}|y_i - y_{i-m}|}\right| \times 100\ \%. \tag{9.57}$$

This metric is the relative MAE scaled with the MAE obtained during training for a naïve forecast[§], with O_t representing the number of observations used for training. For non-seasonal data, $m = 1$, while for seasonal data, m represents the period in which the season occurs and, in this case, the error of the seasonal naïve forecast is used instead.

As mentioned by Hyndman and Athanasopoulos (2013), the main drawback of MAPE and MASE metrics is the fact of being infinite or undefined if $y_i = 0$ and having extreme values when any y_i is close to zero.

[§]Naïve forecast models are models in which the future value to predict is equal to the last observed value (or value observed at the same time in the previous period, in case of seasonal data). Any developed forecasting model should present a better or at least an equal performance compared to the naïve model.

10. Energy recovery using hydroturbines

The main necessary conditions for the installation of small-scale hydropower schemes are presented. Methods to select the most appropriate location for the installation of hydropower schemes in water supply networks are addressed, as well as methods for the selection of the most adequate turbine for a specific site, including a preliminary financial analysis.

Among several renewable energy alternatives in water supply systems (WSS), the installation of micro hydroelectric plants has standing out. In hydropower systems it is usual the use of turbines or pumps operating as turbines (PATs) for the recovery of the excess of energy that is generally lost in the WSS due to the use of pressure reduction valves (PRVs). Such turbo machines extract the potential energy from the fluid (water) and convert it into useful work. However, the selection of the most adequate and profitable site in a network for the installation of a certain type of turbine is generally not a simple task due to the complexity of the pipe system and the variability of the conditions for hydropower generation.

10.1 Conditions for hydropower generation

According to the principle of conservation of energy already presented in Chapter 7.1 (Equation 7.3), the energy balance of a steady flow from A to B will obey to the following relationship (Ramos et al., 2000):

$$Z_A + \frac{p_A}{\gamma} + \frac{v_A^2}{2g} = Z_B + \frac{p_B}{\gamma} + \frac{v_B^2}{2g} + \Delta H_{AB}, \tag{10.1}$$

where ΔH_{AB} corresponds to the headloss between A and B and is equal to the difference between the total heads at A, H_A, and at B , H_B. This available head difference between A and B ($H_A - H_B$), also called gross head, H_{gross}, can be converted into mechanical and electrical energy using a turbine or PAT. The final useful head or net head H_{net} is smaller than the gross head and depends on the turbo-machinery efficiency, accordingly:

$$H_{net} = \eta_{turb}\eta_{transf}\eta_{gen}\eta_{gear}H_{gross} = \eta_t H_{gross}, \tag{10.2}$$

where η_{turb}, η_{transf}, η_{gen} and η_{gear} are the turbine, transformer, generator and gearbox efficiency, respectively. The turbine and generator are the primary mechanical and electrical components of a

small-scale hydropower plant (Natural Resources Canada, 2005).

Not only adequate head but also adequate flow are necessary requirements for hydropower generation (Ramos et al., 2000). Once a flow rate and head has been estimated, the following equation can be used to estimate the capacity of a hydropower plant:

$$P_{\mathrm{d}} = \gamma Q_{\mathrm{d}} H_{\mathrm{d}} \eta_{\mathrm{t}} \tag{10.3}$$

where P_{d} represents the power to be installed in kW, $\gamma = \rho g$ (kN/m^3) where ρ is the density of the water and g is the gravitational acceleration, Q_{d} is the turbine design flow, H_{d} is the turbine design head, or gross head (m), and η_{t} is the efficiency of the set turbine, generator, transformer and gearbox, if applicable. Efficiency is evaluated after the plant configuration is finalised and the turbine selected (Colorado Energy Office, 2015).

Internationally, small-scale hydropower schemes are frequently classified according to their installed power capacity: micro (under 100 kW), mini (100 kW to 1 MW) or small (1 MW to 50 MW) (Natural Resources Canada, 2005). Due to a non-universally accepted definition in EU member states to the definition of small hydropower, the European Small Hydropower Association (2004a) adopted 10 MW as the upper limit for installed capacity of small-scale hydropower plants. According to the head, the European Small Hydropower Association (2004a) classify the hydropower schemes in three categories: low head (2 to 30 m), medium head (30 to 100 m) and high head (equal or above 100 m).

There are also different type of schemes for hydropower generation (European Small Hydropower Association, 2004a): (i) run-of-river schemes, (ii) schemes with the powerhouse located at the base of a dam and (iii) schemes integrated on a canal or in a water supply pipe, which is the approach addressed in this thesis.

Small hydropower plants are less complex than the large ones due to the possibility of integrating the hydraulic conveyance circuit in other components for multiple purposes (such as water supply pipes) (Ramos et al., 2000). Figure 10.1 provides an example demonstrating the integration of a hydropower scheme in a water supply system by replacing an existent pressure reducing valve (PRV). In this type of scheme, a bypass valve system should always be installed to ensure the supply of the water in case of turbine failure (European Small Hydropower Association, 2004a).

10.2 Sites location methods

Potential sites for energy recovery in water supply networks correspond to the sites presenting the highest values of potential recoverable power without compromising the pressure requirements in the supply zones. Since the power is dependent on the available flow and head ranges, the determination of such ranges in each pipe of the network should be the first step for the identification of a potential site for energy recovery.

Typically, in the traditional run-of-river schemes, a single value for the gross head associated to differences in elevation is used as the design head. However, in water networks, the value of head drop (gross head) in each location can be highly variable due to the variation of water levels in reservoirs. An average value of head can be considered as the design head for a preliminary assessment of a potential site. However, it should be taken into account that the determination of an average head may require some understanding on the effects of variation in head on the annual energy production

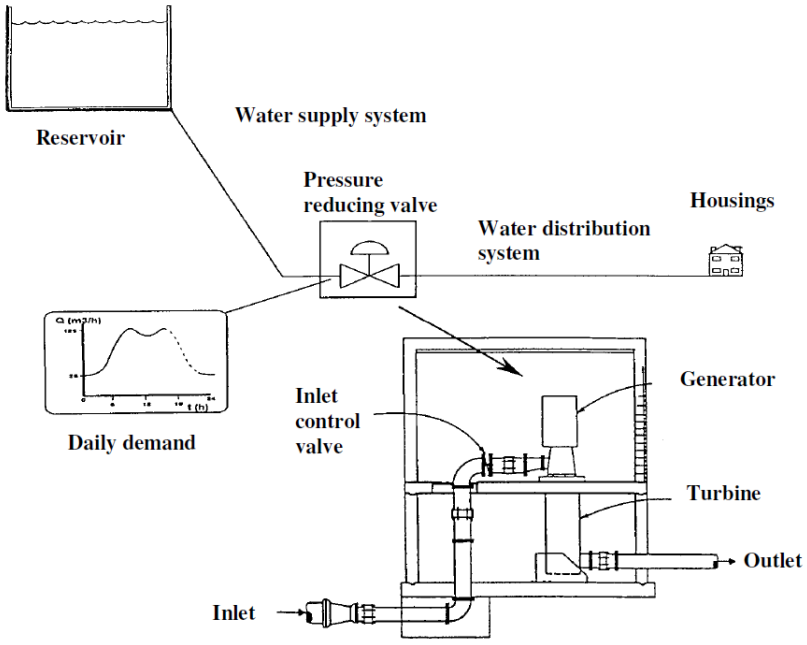


Figure 10.1: Example of a possible application of a micro-turbine in a water network by replacing a pressure reducing valve (Ramos et al., 2000).

(Natural Resources Canada, 2005).

For the estimation of the flow conditions in potential sites in order to determine the design flow, the representation of flow-duration curves (FDC) is a common practice (European Small Hydropower Association, 2004a; Natural Resources Canada, 2005). A FDC is a representation of the historical mean daily flow data series recorded for a number of years (10 years are typically recommended) that facilitates the selection of adequate turbines (Colorado Energy Office, 2015). This type of representation provides, for each recorded value of mean daily flow, the percentage of time such value was equalled or exceeded, *i.e.* provides the number of days that presented the same value of mean flow during the period of recorded data. Examples of FDCs are provided in Figure 10.2.

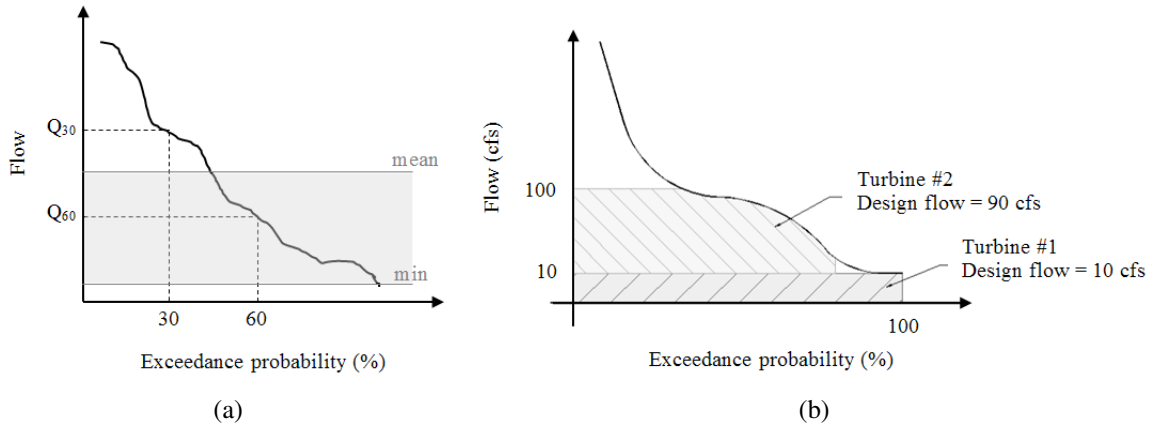


Figure 10.2: (a) Example of a flow-duration curve (FDC) with representation of a possible range of turbine operation and (b) another example showing parallel turbines combination for achieving a desired design flow (adapted from Colorado Energy Office, 2015).

For each recorded value of flow, the corresponding exceedance probability (%) can be calculated by (CDFW, 2013):

$$\chi_{\text{exceed}} = [R/(n_{\text{days}} + 1)] \times 100, \tag{10.4}$$

where χ_{exceed} is the probability that a given flow will be equalled or exceeded, R is the assigned rank number (the flow data is usually sorted from the highest to the lowest value and then a rank is assigned) and n_{days} is the total number of days in the period of record.

The FDC enables the assessment of flow variability at a particular site and the determination of an initial design flow for the hydropower scheme. The Colorado Energy Office (2015) recommends an initial estimation of the design flow for a small hydro system equal to a flow presenting an exceedance probability between 30 and 60 %. The design flow is the maximum flow rate that the hydropower scheme should operate and corresponds to the best efficiency operating point of the turbine (Colorado Energy Office, 2015). In Figure 10.2a, a possible range of operation for a turbine under the presented flow conditions is represented by the grey area. Figure 10.2b provides a different example of utilisation of a FDC for a multiple turbines scheme selection. Multiple turbines can be combined to achieve a desired design flow, providing a certain flexibility. This type of scheme can also be adopted in case a standard turbine size, such as a pump-as-turbine (PAT), cannot accommodate the design flow (Colorado Energy Office, 2015).

After determining the design flow, Q_{d}, and design head, H_{d}, for a specific site, an approximation of the potentially recoverable power can then be computed by:

$$P_{\text{gross}} = \gamma Q_{\text{d}} H_{\text{d}}. \tag{10.5}$$

The net recoverable power will be inferior to the available gross power since it depends on the selected turbine and the associated efficiency of the set turbine-generator (and other equipment, if applicable), η_t, accordingly:

$$P_{\text{net}} = P_{\text{gross}} \eta_{\text{t}}. \tag{10.6}$$

The identification of the most adequate site for the development of a hydropower scheme in a water network is commonly treated as an optimisation problem whose main objective is the maximisation of energy production (power generation) and/or the minimisation of the pressures in the network in order to reduce water leakages (Corcoran, McNabola, & Coughlan, 2015; Fecarotta, Aricò, Carravetta, Martino, & Ramos, 2015; Fontana et al., 2011; Giugni, Fontana, & Ranucci, 2013). In fact, both objectives are interconnected since the pressure reduction can be directly related with the energy recovered by a turbine. In any case, a pressure constraint related with limitations of pressure often imposed by the regulator, for security and comfort reasons (Samora, Franca, Schleiss, & Ramos, 2015), should always be taken into account.

10.2.1 Optimal site location approach

Modelling a pressure reducing valve (PRV) in specific locations of the network and obtaining the flow rate and the head drop through the valve during a simulation period allows to compute the energy that is dissipated in such equipment, and so, the potentially recoverable energy. In EPANET, the head drop through a PRV is computed by the equation of the minor headlosses through the link, as presented in

equation 7.5, which can be re-written in terms of the flow rate, Q, and the valve/link diameter, D:

$$h_{\mathrm{m}} = K_{\mathrm{m}} \frac{2}{\pi g} \left(\frac{Q}{D} \right)^2, \tag{10.7}$$

where the minor loss coefficient, K_{m}, is the parameter that can be adjusted by the user in order to maximise the head drop, and thus, maximise the potentially recoverable power.

Similarly with the approach proposed by Giugni et al. (2013), in this work, the determination of optimal locations for a turbine installation is performed using an optimisation approach to find the best locations of PRVs (and the associated loss coefficient) in order to maximise the potentially recoverable energy. Such optimisation problem can be mathematically described as:

$$\begin{aligned} &\underset{K_{\mathrm{m}}}{\text{maximise}} && E_{\mathrm{recov}} = \gamma \sum_{t} Q_t h_{\mathrm{m},t}(K_{\mathrm{m}}) t_{\mathrm{step}}, && t = 1, \ldots, n_{\mathrm{steps}} \\ &\text{subject to} && P_{i,t} - P_{\mathrm{min}} \geq 0, && i = 1, \ldots, n_{\mathrm{dnodes}}, \end{aligned} \tag{10.8}$$

where t_{step} is the duration of the time-step, n_{dnodes} is the number of demand nodes, P_{min} (m) is the minimum pressure required in the demand nodes and $P_{i,t}$ is the pressure at node i in the time-step t. In this optimisation problem, the decision variable is the minor loss coefficient, K_{m}, which can take values, for instance, as the ones presented in Figure 10.3 according to the valve opening (from 0 - fully closed - to 1 - fully opened).

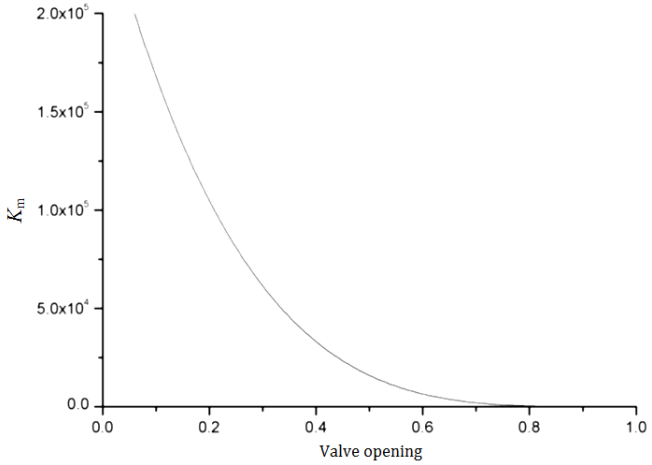


Figure 10.3: Variation of the headloss coefficient of a needle valve with the percent of valve opening (Fontana et al., 2011).

The idea is to test distinct scenarios in a network modelled with EPANET by automatically installing a PRV in each existent pipe at a time (one different scenario for each tested pipe) and determine the valve's minor loss coefficient that maximizes the potentially recoverable energy, E_{recov}.

A simple line search strategy can be used to solve the presented optimisation problem. In this method, the search starts at an arbitrary point K_{m}^0 and then, a certain step size, ϕ_{s}^j determines how far this initial solution should move along a certain direction, d^j until find the optimum (maximum, in this case). Using a line search strategy, any new solution, K_{m}^{j+1}, can be found according to:

$$K_{\mathrm{m}}^{j+1} = K_{\mathrm{m}}^j + \phi_{\mathrm{s}}^j d^j. \tag{10.9}$$

Instead of using the well-known steepest descent method to determine, in each j^{th} iteration, the search direction, d^j, the fixed value of $d^j = 1$ can be used. The initial solution considered is $K_{\text{m}}^0 = 0$, which correspond to a fully opened valve. The solution is set to move along the search space in fixed steps of size $\phi_{\text{s}}^j = 500$. In each iteration, the objective function, E_{recov}, is evaluated and the iterative process stops when its value is maintained or decreases.

10.3 Turbine selection/design methods

The development of a cost-effective and efficient small-scale hydropower project implies the optimal selection/design of the hydroturbine(s) (Sangal, Arpit, & Dinesh, 2013). In small-scale hydraulic applications, the turbines used are scaled-down versions of the conventional large hydroturbines (Natural Resources Canada, 2005). The main types of existing hydroturbines are represented in Figure 10.4. Generally, a turbine can be classified according to the basis of principle of operation as: (i) impulse turbines (Pelton, Cross-flow and Turgo turbines) or (ii) reaction turbines (Francis, Kaplan/propeller and pumps-as-turbines).

Reaction turbines operate with pressurised flow (Ramos et al., 2000). In this type of turbine, part of the water pressure changes as it moves through the turbine since the pressure energy is transformed into mechanical rotational energy of the runner.

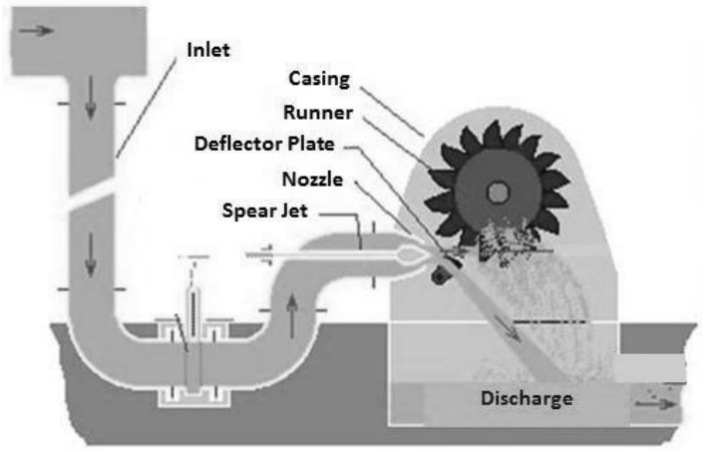
In impulse turbines, the water acts in the runner as a free jet at atmospheric pressure. The kinetic energy available in the water that comes from the nozzle(s) is transformed into rotational mechanical energy when the water jet hits the blades of the runner (Ramos et al., 2000).

The optimal selection/design of a turbine should take into account both technical parameters, such as the specific speed, diameter and efficiency of the turbine, and financial parameters, such as the costs of equipment and costs of installation, costs for civil works, etc. Not only the technical but also the financial viability of each potential small-scale hydropower project are very site specific (Natural Resources Canada, 2005).

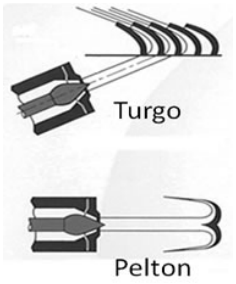
In this work, the methods presented to select/design turbines and evaluate the corresponding technical and financial feasibility are essentially based on the approaches followed by the RETScreen® software for micro hydropower plants. RETScreen® is a clean energy project analysis software tool that allows to determine the technical and financial viability of potential renewable energy, energy efficiency and co-generation projects (Natural Resources Canada, 2005). This tool contains an International Small Hydro Project Model that can be used world-wide to evaluate the energy production, life-cycle costs and greenhouse gas emissions reduction for small-scale hydropower schemes, ranging in size from multi-turbine small and mini hydro installations to single-turbine micro hydro systems (Natural Resources Canada, 2005).

10.3.1 Technical feasibility

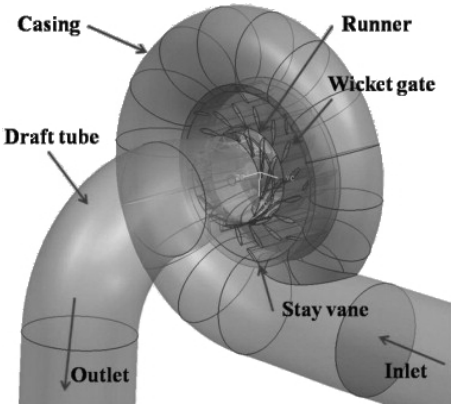
In technical terms, the selection of an appropriate type of turbine should be based on its suitability to the available head and flow in the site for the proposed hydropower plant(Natural Resources Canada, 2005). Figure 10.5 shows the operating ranges of the main existent turbines. As stated by the Colorado Energy Office (2015), a preliminary use of this type of chart enables the identification of potential turbine types that are suitable for a given design head and flow.



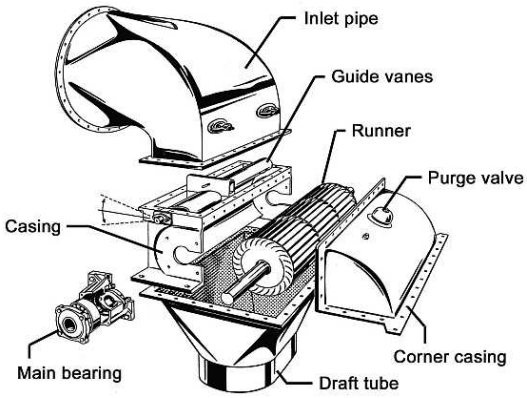
(a)



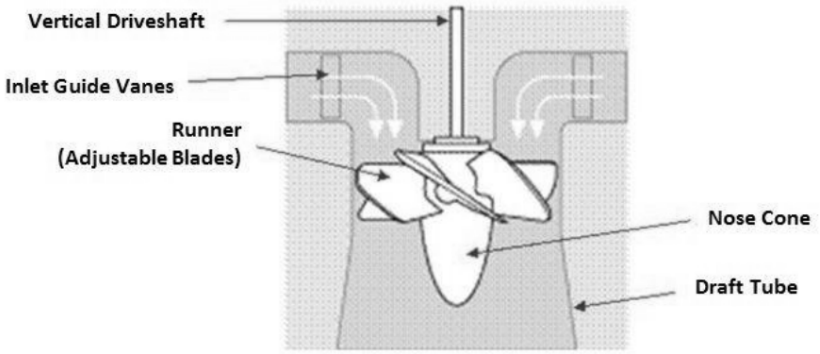
(b)



(c)



(d)



(e)

Figure 10.4: Representation of the main types of hydroturbines: (a) Pelton-type turbine, (b) difference in the interaction of a water jet coming from the nozzle and hitting the blades of the runner of a Pelton- and Turgo-type turbine, (c) Francis-type turbine, (d) Cross-flow turbine and (e) Kaplan-type turbine (adapted from Colorado Energy Office, 2015).

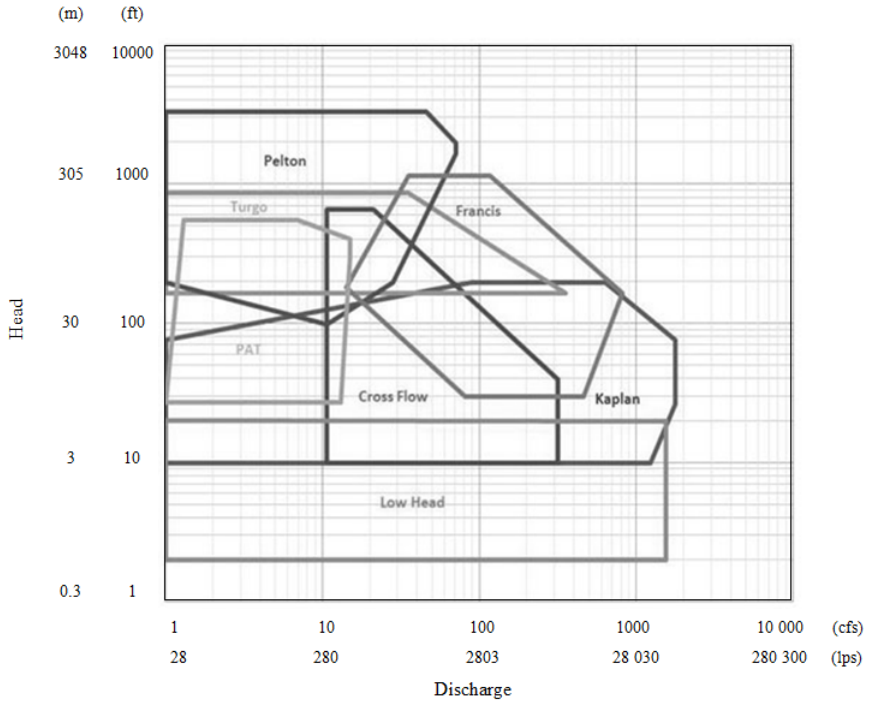


Figure 10.5: Operating ranges of the main existent turbines for small hydropower generation (adapted from Colorado Energy Office, 2015).

The head range should be the first criterion to take into account in the turbine's selection (European Small Hydropower Association, 2004b). Table 10.1 shows the operation head ranges for each main type of turbine, as specified by the European Small Hydropower Association (2004b). As can be observed, more than one type of turbine can be used for some head ranges.

Table 10.1: Operation range for the main existent hydroturbines in terms of head (European Small Hydropower Association, 2004b).

Operation range	Turbine type	
50<H<1300	Pelton	impulse
50<H<250	Turgo	impulse
10<H<350	Francis	reaction
3<H<250	Cross-flow	impulse
2<H<40	Kaplan/propeller	reaction

In the process of selection of a turbine, the associated efficiency of the turbo-machine also presents an important role since the net recovered energy will be dependent of such parameter. Any turbine has an associated efficiency curve as a function of the flow. Turbines that present high efficiencies under broad ranges of flow are adequate for schemes developed in water networks due to the high variability of flow in such systems. However, turbines capable of covering a large operating range are also typically more expensive (Colorado Energy Office, 2015). Figure 10.6 depicts the variation on the efficiency of the main types of turbine under the operation at flow rates distinct from the design flow.

The turbine's efficiency curves take into account a number of factors including the rated head,

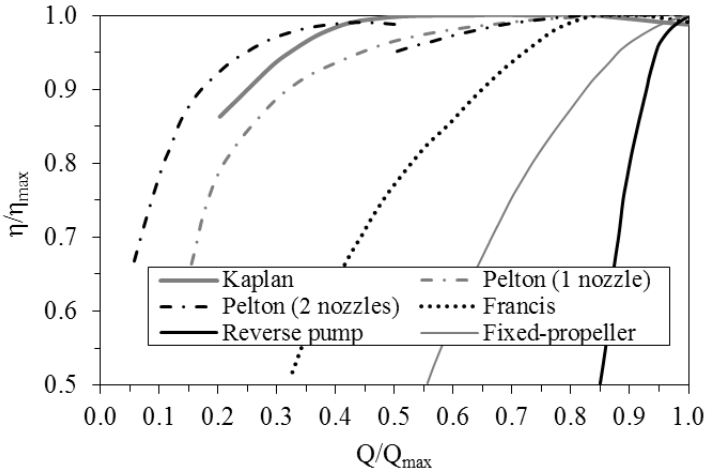


Figure 10.6: Relative efficiencies regarding the discharges as proportion of design flow rate for Pelton (1 and 2 nozzles), Kaplan, propeller with fixed guide vanes and blades and Francis turbines and reverse pump (or pump-as-turbine) (adapted from Aline et al., 2012; European Small Hydropower Association, 2004b).

the runner diameter and the turbine specific speed (Natural Resources Canada, 2005). In this work, the determination of each turbine efficiency, based on its specific speed (if applicable) and runner diameter, which are also designed parameters, is performed using the equations proposed by the Natural Resources Canada (2005). Such equations were derived from a large number of manufacture efficiency curves for different turbine types and head and flow conditions.

For reaction turbines (Francis, Kaplan and propeller), the runner diameter, D_t, can be determined by (Natural Resources Canada, 2005):

$$D_t = k_1 Q_d^{0.473}, \tag{10.10}$$

where Q_d is the design flow rate (m^3/s) and $k_1 = 0.46$. The dimensionless specific speed based on flow, N_Q, is obtained by:

$$N_Q = k_2 H_t^{-0.5}, \tag{10.11}$$

where H_t (m) is the rated head on turbine and the constant k_2 takes the value 600 for Francis turbines and 800 for Kaplan or propeller turbines.

After calculating the diameter and speed, the efficiency for each distinct type of reaction turbine can be computed according to the formulae presented as following.

In the case of Francis turbines, the peak efficiency can be determined by (Natural Resources Canada, 2005):

$$\eta_{F,\,peak} = \left[0.919 - \left(\frac{N_Q - 56}{256}\right)^2 + \left(0.81 + \left(\frac{N_Q - 56}{256}\right)^2\right)\left(1 - \frac{0.789}{D_t^{0.2}}\right)\right] - 0.0305 + 0.005 R_m, \tag{10.12}$$

where R_m is the turbine manufacture/design coefficient that can take values from 2.8 to 6.1 (the default value considered in this work is 4.5). In case of flows, Q_t, below the peak efficiency flow, Q_{peak}, the

turbine efficiency should be calculated by:

$$\eta_{\text{F,below}} = \left[1 - \left(1.25\left(\frac{Q_{\text{peak}} - Q_{\text{t}}}{Q_{\text{peak}}}\right)^{3.94-0.0195N_Q}\right)\right]\eta_{\text{F, peak}}. \tag{10.13}$$

By other side, for flows above peak efficiency flow, the efficiency equation is:

$$\eta_{\text{F,above}} = \eta_{\text{F, peak}} - \left[\left(\frac{Q_{\text{t}} - Q_{\text{peak}}}{Q_{\text{d}} - Q_{\text{peak}}}\right)^2 \left(\eta_{\text{F, peak}} - (1 - 0.0072N_Q^{0.4})\eta_{\text{F, peak}}\right)\right]. \tag{10.14}$$

The efficiency of Kaplan turbines is given by (Natural Resources Canada, 2005):

$$\eta_{\text{K}} = \left[1 - 3.5\left(\frac{Q_{\text{peak}} - Q_{\text{t}}}{Q_{\text{peak}}}\right)^6\right]\eta_{\text{K,peak}}, \tag{10.15}$$

where $Q_{\text{peak}} = 0.75Q_{\text{d}}$ and

$$\eta_{\text{K,peak}} = 0.905 - \left(\frac{N_Q - 170}{700}\right)^2 + \left(0.095 + \left(\frac{N_Q - 170}{700}\right)^2\right)\left(1 - \frac{0.789}{D_{\text{t}}^{0.2}}\right) - 0.0305 + 0.005R_{\text{m}}. \tag{10.16}$$

For propeller turbines, the efficiency computation is performed by (Natural Resources Canada, 2005):

$$\eta_{\text{prop}} = \left[1 - 1.25\left(\frac{Q_{\text{peak}} - Q_{\text{t}}}{Q_{\text{peak}}}\right)^{1.13}\right]\eta_{\text{prop,peak}}, \tag{10.17}$$

where $Q_{\text{peak}} = Q_{\text{d}}$ and $\eta_{\text{prop,peak}} = \eta_{\text{K,peak}}$, which can be obtained from Equation 10.16.

For impulse turbines (Pelton, Turgo and cross-flow), the specific speed is not determined. The equation that gives the efficiency of Pelton turbines is (Natural Resources Canada, 2005):

$$\eta_{\text{P}} = \left[1 - (1.31 + 0.025n_{\text{jet}})\left|\frac{Q_{\text{peak}} - Q_{\text{t}}}{Q_{\text{peak}}}\right|^{(5.6+0.4n_{\text{jet}})}\right]\eta_{\text{P,peak}}, \tag{10.18}$$

where n_{jet} is the number of jets (value from 1 to 6), $Q_{\text{peak}} = (0.662 + 0.001n_{\text{jet}})Q_{\text{d}}$ and $\eta_{\text{P,peak}}$ can be calculated by:

$$\eta_{\text{P,peak}} = 0.864D_{\text{out}}^{0.04}, \tag{10.19}$$

where the outside runner diameter, D_{out}, is given by:

$$D_{\text{out}} = \frac{49.4H_{\text{t}}^{0.5}n_{\text{jet}}^{0.02}}{31\left[H_{\text{t}}(Q_d/n_{\text{jet}})\right]^{0.5}}, \tag{10.20}$$

where H_{t} (m) is the head available in the turbine.

The efficiency of Turgo turbines can be considered as 30 % of the efficiency of Pelton turbines (Natural Resources Canada, 2005):

$$\eta_{\text{T}} = 0.3\eta_{\text{P}}. \tag{10.21}$$

Finally, the determination of the efficiency of cross-flow turbines is performed by (Natural Resources Canada, 2005):

$$\eta_{\mathrm{CF}} = 0.79 - 0.15\left(\frac{Q_d - Q_{\mathrm{t}}}{Q_{\mathrm{peak}}}\right) - 1.37\left(\frac{Q_d - Q_{\mathrm{t}}}{Q_{\mathrm{peak}}}\right)^{14}, \tag{10.22}$$

where $Q_{\mathrm{peak}} = Q_{\mathrm{d}}$.

Since the flow rate in water supply and distribution systems is usually highly variable, the turbine's efficiency, and hence the recoverable power, will be also largely variable.

10.3.2 Financial feasibility

The idea of a financial feasibility analysis is to determine whether the balance of costs and savings of a certain project is attractive (Natural Resources Canada, 2005).

The financial analysis model proposed by the Natural Resources Canada (2005) uses formulae based on standard financial terminology that can be found in most financial textbooks. The model considers year 0 as the initial investment year and that the timing of cash flows occurs at the end of the year.

For a complete financial feasibility study, a large number of contributors for the initial costs of a small-scale hydropower project, such as (i) development, (ii) engineering, (iii) energy equipment and installation, (iv) access road, (v) transmission line, (vi) transformer and installation, (vii) civil works, (viii) penstock and installation, (ix) canal and (x) others (miscellaneous), is included in the model proposed by the Natural Resources Canada (2005). However, in the proposed preliminary financial analysis, only the energy equipment, its installation and civil works are considered in the costs computation for the comparison among different scenarios. The total investment is then obtained by summing such costs.

Following the methodology of the Natural Resources Canada (2005), the turbines costs can be estimated by:

$$C_{\mathrm{t,F}} = 0.17 n_{\mathrm{turb}}^{0.96} J_{\mathrm{t}} K_{\mathrm{t}} D_{\mathrm{a}}^{1.47}\left[(13 + 0.01H_{\mathrm{d}})^{0.3} + 3\right] 10^6, \quad \text{for Francis turbines,} \tag{10.23a}$$

$$C_{\mathrm{t,K}} = 0.27 n_{\mathrm{turb}}^{0.96} J_{\mathrm{t}} K_{\mathrm{t}} D_{\mathrm{a}}^{1.47}\left(1.17H_{\mathrm{d}}^{0.12} + 2\right) 10^6, \quad \text{for Kaplan turbines,} \tag{10.23b}$$

$$C_{\mathrm{t,prop}} = 0.125 n_{\mathrm{turb}}^{0.96} J_{\mathrm{t}} K_{\mathrm{t}} D_{\mathrm{a}}^{1.47}\left(1.17H_{\mathrm{d}}^{0.12} + 4\right) 10^6, \quad \text{for propeller turbines,} \tag{10.23c}$$

$$C_{\mathrm{t,P/T}} = \begin{cases} 3.47 n_{\mathrm{turb}}^{0.96}\left(\frac{P_{\mathrm{u}}}{H_{\mathrm{d}}^{0.5}}\right)^{0.44} 10^6, & \text{if } \frac{P_{\mathrm{u}}}{H_{\mathrm{d}}^{0.5}} > 0.4 \\ 5.34 n_{\mathrm{turb}}^{0.96}\left(\frac{P_{\mathrm{u}}}{H_{\mathrm{d}}^{0.5}}\right)^{0.91} 10^6, & \text{if } \frac{P_{\mathrm{u}}}{H_{\mathrm{d}}^{0.5}} \leq 0.4 \end{cases}, \quad \text{for Pelton/Turgo turbines,} \tag{10.23d}$$

and

$$C_{\mathrm{t,CF}} = 0.5 C_{\mathrm{t,P/T}}, \quad \text{for cross-flow turbines.} \tag{10.23e}$$

In Equations 10.23a to 10.23e, n_{turb} represents the number of turbines, which is always 1 for micro hydroplants, $D_{\mathrm{a}} = 0.482 Q_{\mathrm{d}}^{0.45}$ is the approximated turbine runner diameter (m), J_{t} is an higher cost vertical axis turbine factor to account for cost increase with vertical axis at heads above 25 m ($J_{\mathrm{t}} = 1.1$ if $H_{\mathrm{d}} > 25$ m, otherwise $J_{\mathrm{t}} = 1$), K_{t} is a lower cost small horizontal axis turbine factor to account for cost decrease with small horizontal axis units ($K_{\mathrm{t}} = 0.9$ if $D_{\mathrm{a}} < 1.8$ m, otherwise $K_{\mathrm{t}} = 1$), and

$P_u = 7.53 Q_d H_d / 1000$ is the unit capacity (MW).

For all types of turbine, the cost of installation is considered to be equal to 15 % of the turbine cost (Natural Resources Canada, 2005):

$$C_{instal} = 0.15 C_t \tag{10.24}$$

Costs of civil works (in US$), for micro hydropower projects, are computed by (Natural Resources Canada, 2005):

$$C_{civil} = 10^6 \frac{1.97}{n_{turb}^{0.04}} f_{civil} \left(\frac{P_u}{H_d^{0.3}} \right)^{0.82}, \tag{10.25}$$

where f_{civil} is a civil cost factor (0.44 for an existing dam or 1.0 if no dam exists).

After calculating the total investment cost for the project development, the annual revenue should be estimated by means of the expected annual recovered energy and the sell price of energy. It should be noticed that the sell price of energy as well as the existence of possible incentives are variable factors from country to country and can be determinant in the final decision on the financial feasibility of the project.

To determine the return on investment, *i.e.* the number of years it takes to equal the total investment, a simple cash-flow analysis can be performed. A cash-flow tracks, on an annual basis, all expenses (outflows) and incomes (inflows) generated by the hydropower project. In the expenses, the initial investments and annual operation and maintenance costs should be included. The incomes may include not only the produced energy but also acquired incentives and grants (Natural Resources Canada, 2005).

References

Adamowski, J., & Karapataki, C. (2010). Comparison of multivariate regression and artificial neural networks for peak urban water demand forecasting: evaluation of different ANN learning algorithms. *Journal of Hydrologic Engineering*, *15*(10), 729–743.

Aline, C., Vincent, D., & Petras, P. (2012). Integration of small hydro turbines into existing water infrastructures. In H. Samadi-Boroujeni (Ed.), *Hydropower - practice and application* (chap. 12). Retrieved 2015, from `http://www.intechopen.com/books/hydropower-practice-and-application/integration-of-small-turbines-into-water-infrastructure` doi: 10.5772/35251

Andrade-Campos, A., Dias-de-Oliveira, J., & Pinho-da-Cruz, J. (2015). *Otimização não-linear em engenharia*. ETEP - Edições Técnicas e Profissionais.

Armstrong, J. S. (2001). Evaluating forecasting methods. In *Principles of forecasting: A Handbook for Researchers and Practitioners* (pp. 443–472). Springer.

Atiya, A. (1991). *Learning algorithms for neural networks* (Unpublished doctoral dissertation). California Institute of Technology.

A basic introduction to neural networks. (2015). `http://pages.cs.wisc.edu/~bolo/shipyard/neural/local.html`. (Accessed: July 2015)

Bennett, N. D., Croke, B. F., Guariso, G., Guillaume, J. H., Hamilton, S. H., Jakeman, A. J., ... Andreassian, V. (2013). Characterising performance of environmental models. *Environmental Modelling & Software*, *40*, 1–20.

Caudill, M. (1987). Neural networks primer, part 1. *AI expert*, *2*(12), 46–52.

CDFW. (2013). *Standard operating procedure for flow duration analysis in california.* Retrieved 2016, from `http://www.dfw.ca.gov/water/instream_flow.html` (CDFW-IFP-005)

Coelho, B., & Andrade-Campos, A. (2016). A new approach for the prediction of speed-adjusted pump efficiency curves. *Journal of Hydraulic Research*. (accepted for publication)

Colorado Energy Office. (2015). *The small hydropower handbook*. Retrieved 2015, from `https://www.colorado.gov/pacific/sites/default/files/atoms/files/`

Corcoran, L., McNabola, A., & Coughlan, P. (2015). Optimization of water distribution networks for combined hydropower energy recovery and leakage reduction. *Journal of Water Resources Planning and Management*, *142*(2), 04015045.

Costa, L. (2003). *Algoritmos evolucionários em optimização uni e multi-objectivo* (Unpublished doctoral dissertation). Universidade do Minho.

European Small Hydropower Association. (2004a). Guide on How to Develop a Small Hydropower Plant (Part 1).. Retrieved 2015, from `https://energypedia.info/wiki/File:Part_1_guide_on_how_to_develop_a_small_hydropower_plant-_final1.pdf`

European Small Hydropower Association. (2004b). Guide on How to Develop a Small Hydropower Plant (Part 2).. Retrieved 2015, from https://energypedia.info/images/4/4a/Part_2_guide_on_how_to_develop_a_small_hydropower_plant-_final-21.pdf

Fecarotta, O., Aricò, C., Carravetta, A., Martino, R., & Ramos, H. M. (2015). Hydropower potential in water distribution networks: Pressure control by pats. *Water Resources Management*, *29*(3), 699–714.

Fontana, N., Giugni, M., & Portolano, D. (2011). Losses reduction and energy production in water-distribution networks. *Journal of Water Resources Planning and Management*, *138*(3), 237–244.

Giugni, M., Fontana, N., & Ranucci, A. (2013). Optimal location of prvs and turbines in water distribution systems. *Journal of Water Resources Planning and Management*, *140*(9), 06014004.

Goldberg, D. E., & Deb, K. (1991). A comparative analysis of selection schemes used in genetic algorithms. In G. J. E. Rawlins (Ed.), *Foundations of genetic algorithms* (Vol. 1, pp. 69–93). Morgan Kaufmann Publishers.

Gulich, J. (2003). Effect of reynolds number and surface roughness on the efficiency of centrifugal pumps. *Journal of fluids engineering*, *125*(4), 670–679.

Hyndman, R., & Athanasopoulos, G. (2013). *Forecasting: principles and practice*. Retrieved May, 2015, from http://otexts.org/fpp/

Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. In *Proceedings of ieee international conference on neural networks* (Vol. 4, pp. 1942–1948).

Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, *220*(4598), 671–680.

Luk, K., Ball, J., & Sharma, A. (2000). A study of optimal model lag and spatial inputs to artificial neural network for rainfall forecasting. *Journal of Hydrology*, *227*(1), 56–65.

Makridakis, S., & Hibon, M. (1997). Arma models and the box–jenkins methodology. *Journal of Forecasting*, *16*(3), 147–163.

Marchi, A., & Simpson, A. R. (2013). Correction of the epanet inaccuracy in computing the efficiency of variable speed pumps. *Journal of Water Resources Planning and Management*, *139*(4), 456–459.

Martin, C. S. (2000). *Hydraulic transient design for pipeline systems.* McGraw-Hill, New York.

Mathworks: arima class. (2015). http://www.mathworks.com/help/econ/arima-class.html#bti8g46-6. (Accessed: 2015)

Mathworks: Autocorrelation and partial autocorrelation. (2015). http://www.mathworks.com/help/econ/autocorrelation-and-partial-autocorrelation.html. (Accessed: 2015)

Montgomery, D. C., Jennings, C. L., & Kulahci, M. (2008). *Introduction to time series analysis and forecasting*. John Wiley & Sons.

Morton, W. R. (1975). Economics of ac adjustable speed drives on pumps. *IEEE Transactions on Industry Applications*(3), 282–286.

Natural Resources Canada. (2005). *Clean Energy Project Analysis: RETScreen Engineering & Cases Textbook* (3rd ed.). Retrieved 2015, from http://publications.gc.ca/collections/collection_2007/nrcan-rncan/M154-13-2005E.pdf

Nelder, J. A., & Mead, R. (1965). A simplex method for function minimization. *The computer journal*, *7*(4), 308–313.

Nocedal, J., & Wright, S. (2006). *Numerical optimization*. Springer Science & Business Media.

Quintela, A. C. (1981). *Hidráulica* (2nd ed.). Lisboa: Fundação Calouste Gulbenkian. (in portuguese)

Ramos, H., Almeida, A., Portela, M., & Almeida, H. P. (2000). *Guidelines for design of small hydropower plants*. WREAN (Western Regional Energy Agency and Network) and DED (Department of Economic Development-Energy Division).

Rao, S. S. (2009). *Engineering optimization: theory and practice*. John Wiley & Sons.

Rojas, R. (1996). *Neural Networks: A Systematic Introduction*. Springer-Verlag.

Rossman, L. A. (2000). *Epanet 2: users manual*. US Environmental Protection Agency. Office of Research and Development. National Risk Management Research Laboratory.

Samora, I. A., Franca, M. J., Schleiss, A., & Ramos, H. (2015). Optimal location of micro-turbines in water supply network. In *Proceedings of 36th iahr world congress 2015.*

Sangal, S., Arpit, G., & Dinesh, K. (2013). Review of optimal selection of turbines for hydroelectric projects. *International Journal of Emerging Technology and Advance Engineering*, *3*, 424–430.

Sárbu, I., & Borza, I. (1998). Energetic optimization of water pumping in distribution systems. *Mechanical Engineering*, *42*(2), 141–152.

Simpson, A. R., & Marchi, A. (2013). Evaluating the approximation of the affinity laws and improving the efficiency estimate for variable speed pumps. *Journal of Hydraulic Engineering*, *139*(12), 1314–1317.

Sitzenfrei, R., Berger, D., & Rauch, W. (2015). Design and optimization of small hydropower systems in water distribution networks under consideration of rehabilitation measures. *Urban Water Journal*, 1–9.

Smith, A. E., & Coit, D. W. (1997). Constraint-handling techniques - penalty functions. In *Handbook of evolutionary computation* (chap. C5.2). Institute of Physics Publishing and Oxford University Press.

Solomatine, D. P., & Siek, M. B. (2006). Modular learning models in forecasting natural phenomena. *Neural networks*, *19*(2), 215–224.

Stat 510 – applied time series analysis. (2015). `https://onlinecourses.science.psu.edu/stat510/node/62`. (Accessed: 2015)

Storn, R., & Price, K. (1995). *Differential evolution - a simple and efficient adaptive scheme for global optimization over continuous spaces* (Vol. 3). International Computer Science Institute, Berkeley.

US EPA: EPANET. (2015). `http://www2.epa.gov/water-research/epanet`. (Accessed: 2015)

Valente, R. A., Andrade-Campos, A., Carvalho, J. F., & Cruz, P. S. (2011). Parameter identification and shape optimization. *Optimization and Engineering*, *12*(1-2), 129–152.

Van Zyl, J. E., Savic, D. A., & Walters, G. A. (2004). Operational optimization of water distribution systems using a hybrid genetic algorithm. *Journal of Water Resources Planning and Management*, *130*, 160-170.

Vieira, F., & Ramos, H. M. (2009). Optimization of operational planning for wind/hydro hybrid water supply systems. *Renewable Energy*, *34*(3), 928–936.

Viessman, W., Hammer, M. J., Perez, E. M., & Chadik, P. A. (2009). *Water supply and pollution control*. Pearson Prentice Hall New Jersey, NJ.

Walski, T., Chase, D., & Savic, D. (2001). *Water distribution modeling*. Haestad Press.

Walski, T., Zimmerman, K., Dudinyak, M., & Dileepkumar, P. (2003). Some surprises in estimating the efficiency of variable-speed pumps with the pump affinity laws. In *World water and environmental resources congress.*

Yang, X.-S. (2010). *Engineering optimization: an introduction with metaheuristic applications*. John Wiley & Sons.

Part IV

Implementation

11. Integrated computational tool

A general overview of the developed numerical tool is provided, describing the main process performed in each module that constitutes the developed tool as well as all connections between them.

In order to obtain an automatic process for the efficiency improvement of water supply systems (WSS) in any kind of network configuration, an integrated numerical tool was developed in C/C++ programming language. The hydraulic simulator EPANET 2.0 was integrated in the developed tool in order to perform the hydraulic simulation of the networks' operation. The incorporation of EPANET in the developed application was performed using the EPANET programming toolkit, which consists in a dynamic link library (DLL) of functions that allow developers to customise EPANET to their own needs (EPA U.S., 2015). It is particularly used in iterative processes (such as design optimisation, operational optimisation, calibration).

A flowchart representing the main processes that take place in the developed tool is presented in Figure 11.1. The numerical tool is composed of four main modules: (i) simulation, (ii) optimisation, (iii) energy recovery and (iv) forecasting, from which only the three first modules were already fully automatically integrated. The forecasting module is an independent tool that provides updated information for the other modules, particularly for the simulation module.

The numerical tool has an initial stage of data collection concerning the user's choices and the model of the network to be optimised. After this initial stage, depending on the user's decision, three distinct processes may be performed:

- Optimisation of the network operation in order to minimise the pumping energy consumption;
- Network analysis through the simulation considering, for instance, distinct methods for computation of the variable-speed pumps efficiency and saving the simulation results in specific formats;
- Search for possible locations for turbines installation in order to recover dissipated energy (from zones with excessive pressures) and selection of the most adequate turbine.

Before start using the presented tool, the EPANET model of a certain network (or section of a network) needs to be configured by the user. The idea is to use such model to obtain an input file, named <network>.inp (see Figure 11.1), where <network> is the network's name chosen by the

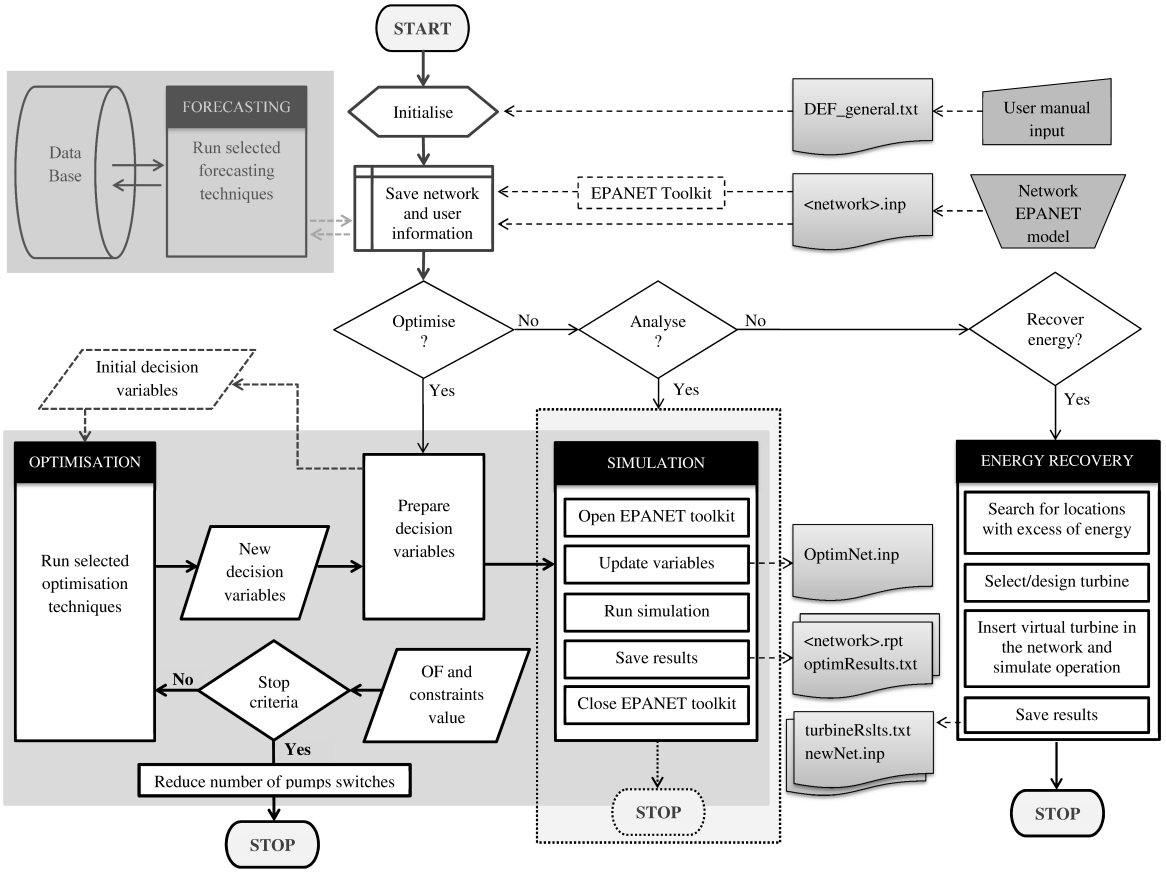


Figure 11.1: Flowchart of the numerical tool developed in this work.

user. In such file, all the characteristics of the network as well as the time parameters for the desired simulation, *i.e.* the simulation duration and size of the time-steps (*e.g.* a simulation of one day performed in hourly time-steps, which gives 24 steps), are included. An EPANET model of a network can be configured by two distinct ways: (i) manually designing the network using the EPANET graphical user interface (GUI) or (ii) converting a CAD* file or a file from a GIS† using file converters such as, for instance, DXF2EPA, EPACAD and epa2gis (see Salomons (2015a, 2015b) and Zonum Solutions (2009)) and importing it to EPANET. The first model configuration option is the most simple and more useful especially for small networks. The option of converting CAD or GIS files into files compatible with EPANET (an ASCII file with extension inp) may be advantageous especially for very large networks (*e.g.* a network of an entire city) but, in turn, requires particular precautions since only pipes and nodes are usually possible to convert. This means that other elements (such as valves and pumps) may need to be configured using the EPANET GUI. Anyway, after importing the files to EPANET, the network layout must always be carefully checked.

Besides the file containing the network EPANET model, another input file is needed to run the presented tool. Such file, named DEF_general.txt (see Figure 11.1), contains several options to define specific characteristics of the processes that may occur. Such options, initially defined by the user,

*CAD is an image file format that allows 2D and 3D designs. CAD files hold information for these images, as well as drafting information. CAD stands for Computer Aided Design.

†GIS stands for Geographical Information System.

include:

File name: the name of the network model, without including the file extension, *i.e.* the previously mentioned <network>.

Switch decisions: include decisions related with (i) the optimisation variables preparation, (ii) the desired water levels in tanks in the end of the defined simulation period, (iii) the method(s) for computing the objective function, *i.e.* the energy costs associated to the network operation, and (iv) the method for the variable-speed pumps efficiency computation. The possible options concerning the first decision are related with the optimisation procedure and, consequently, are described with detail in Section 11.2. The options for the other three decisions are described in the chapter devoted to the networks simulation module (Section 11.1).

Penalty coefficients: the value of the coefficient for each considered penalty. The implementation of penalty methods, related with the optimisation procedure, is described in Section 11.2.

Operation boundaries: the minimum and maximum values to be considered during the optimisation process for (i) the pumps speed, (ii) the levels of water in tanks and (iii) the nodal pressures. For the pumps speed, used as decision variables, the limit values to be introduced by the user correspond to relative speeds (a speed with relation to the nominal speed). For the tanks, the boundary levels are introduced in percentage, *i.e.* the considered minimum and maximum levels correspond, respectively, to a percentage below and above the levels defined for each tank in the model of the network (<network>.inp). Concerning the pressure, only a minimum value is introduced by the user, which avoids unacceptable low pressure in all junctions. Both the tanks water levels and the nodal pressures are controlled through constraint functions during the optimisation procedure. A minimum value of pressure is also used for the energy recovery module.

Optimisation technique: the sequence (and size of such cascade) of the optimisation algorithms to be used. The allowed optimisation techniques are described in Section 11.2.

Elements to optimise: the pumps and valves existent in the network that the user intends to optimise. Only the selected elements will be considered in the optimisation procedure, which means that the control solution for the other elements will remain the same defined in the network model. For both pumps and valves, the user must introduce the numbers 1 or 0 (optimise or not), according to the order they appear in the <network>.inp file.

Optimisation parameters: the parameters that need to be defined for each optimisation algorithm, such as the maximum number of iterations and/or the population size (*i.e.* number of individuals or particles).

In case some of the previous options inserted by the user contain invalid values, an error message with instructions is provided.

This tool was developed in a way to allow an easy evolution to a software with a graphical user interface. The file containing the user options can be simply replaced by boxes to be filled by the users and by buttons, so the users can choose between the distinct available options.

In the beginning of the implemented numerical tool (the starting point in Figure 11.1), an initialisation/preparation process is performed, *i.e.* the programme saves memory for all the processes and initialises distinct structures for each type of element/component that constitute a network (junctions, reservoirs, tanks, pipes, valves and pumps) as well as for the simulation data. The information contained in the file of the network model is read using the functions of the EPANET toolkit

(López-Ibáñez, 2015). Although the toolkit has several functions for retrieving and setting parameters (*ENget*() and *ENset*() functions) that define both the design and the operation of the network, such functions do not allow to retrieve all the information contained in the network model. Thus, new functions (subroutines/methods) were developed in order to read the missing information directly from the file. It was the case of (i) the ID labels for the pumps speed patterns and energy patterns and for the pumps head curves and efficiency curves, (ii) the minimum and maximum operating or alarm levels of each tank and (iii) the value of the demand charge for the computation of additional costs due to maximum power usage by pumps.

At this stage, with the incorporation of a forecasting module in the developed tool, the demand patterns associated to each consumption nodes of the network could be optionally updated according to the predicted values for a near future (24 hours, for instance). This procedure allows obtaining, in advance, the optimal network control decisions for the same period of predicted water demand.

Considering the current development stage of the presented numerical tool, the user has available two possible operations for improving the efficiency of the modelled network: (i) the optimisation of the network operation for the energy costs minimisation or (ii) the search for possible locations of the network presenting excessive pressures and potential for producing energy using hydroturbines. It is also possible to use uniquely the simulation module of the presented tool, which provides additional features to the ones provided by the simulator EPANET 2.0, allowing to perform specific analysis of the networks.

The optimisation module is in fact coupled with the simulation module. The latter is used in the optimisation procedure as a kind of black-box that provides the value of the objective function (OF), as well as the value of the constraint functions, given a certain solution (decision variables) provided by the optimiser, such as demonstrated in Figure 11.1. The computation of the OF is the result of an hydraulic simulation of the network under the new operational conditions provided by the optimiser and computing the associated energy costs.

11.1 Network simulation module

As mentioned before, the simulation module can be used both (i) as a complement of the optimisation procedure or (ii) as a tool for specific analysis of water supply systems.

Some features were implemented in this module, in addition to the ones provided by EPANET, in order to improve the results obtained from the hydraulic simulation of the networks and the use of such results. The main incorporated features include:

- Option for the energy costs computation;
- Distinct methods for the computation of speed-adjusted pump efficiency;
- Save different simulation results in distinct formats in order to assist in the results' analyses;

All the additional features were implemented externally to the EPANET toolkit, *i.e.* no changes were performed in the toolkit and the implementation of new developed functions/subroutines was performed through the direct interaction with the files generated during the simulation process. For this reason, depending on the user's options, the EPANET toolkit may need to be open and closed more than once during the simulation procedure. This implementation strategy requires some additional process time (CPU time) due to the initialisation of the toolkit several times. However, besides being the most simple strategy, it is advantageous in case of (i) using new versions and releases of

the EPANET toolkit are released and (ii) some other researchers, using or not other programming language, want to implement specific features relevant for their works by following the description of the functions/subroutines implemented in this work.

When used within the optimisation module, the simulation procedure corresponds to the evaluation of the objective function. The developed function to implement such procedure is called *evaluateOF* and the main implemented steps are provided in Algorithm 6.

Algorithm 6 Main steps for the objective function evaluation.

```
1:  function EVALUATEOF()
2:      Open toolkit: ENopen(<network>.inp);
3:      switch aggregDecision do
4:          case true
5:              Disaggregate variables obtained from the optimiser: disaggregateXopt();
6:          case false
7:              Save decision variables (obtained from the optimiser): saveXopt();
8:      for All variable-speed pumps to be optimised do
9:          Speed variables update in patterns: ENsetpattern();
10:     end for
11:     for All pumps and valves to be optimised do
12:         Speed/status and time variables update in control statements: ENsetcontrol();
13:     end for
14:     Save updated model: ENsaveinpfile(OptimNet.inp);
15:     Close toolkit: ENclose();
16:     for All fixed-speed pumps and valves to optimise do
17:         Correction of saved controls status: correctControls();
18:     end for
19:     Open toolkit: ENopen(OptimNet.inp);
20:     switch OFdecision do
21:         case OFdecision = epanet
22:             Ojective function value (OF) obtained by EPANET: OFepanet();
23:         case OFdecision = compute
24:             Objective function computed: OFcompute();
25:         case OFdecision = both
26:             Objective function computed and compared with the obtained by EPANET;
27:     Close toolkit: ENclose();
28:     return OF result.
29: end function
```

To evaluate the objective function, the toolkit has to be open/closed at least twice (steps 2/15 and 19/27 in Algorithm 6). First, the decision variables that are provided by the optimisation module (optimiser) need to be introduced (updated) in the initial model of the network (<network>.inp). To keep the initial network unchanged, a network model with the updated variables is saved to a new file (OptimNet.inp). Small corrections need to be performed to this new file, since the toolkit does not save the file exactly in the same format as the initial file. In this particular case, the status in the controls of valves and fixed-speed pumps, that were update with OPEN/CLOSED statements, are saved by the toolkit as 1.0/0.0. Thus, the developed function *correctControls*() (step 17) performs the

"correction" of such statements by (i) opening the file (OptimNet.inp), (ii) replacing the corresponding values and (iii) closing the file. After that, the toolkit is open again with the updated model of the network (OptimNet.inp) in order to proceed with the evaluation of the objective function (using the method initially defined by the user).

The switch *aggregDecision* (explained in Section 11.2) provides an option of reducing the number of decision variables (by aggregation) before initialising the optimisation procedure, in order to speed-up the process. In case this option is selected ($aggregDecision = true$), the inverse process (disaggregation) need to be performed before the simulation process for the solution evaluation.

The switch $OFdecision$ allows the user to choose between the energy costs computation by EPANET ($OFdecision$ =epanet) or, alternatively, compute the costs externally to EPANET ($OFdecision$ = compute), using information previously retrieved from the hydraulic simulation and saved in memory. The former option was initially implemented to overcome a limitation in the use of the toolkit that was limiting the number of iterations (maximum of 65535 iterations, to be more precise[‡]) due to the generation of a temporary file containing the hydraulic results obtained at every time step (a binary hydraulics file with extension *.hyd). Such limitation was then overcome by running the EPANET hydraulic simulation without saving the hydraulic results to files and using the required information saved in memory for the costs computation ($OFcompute()$ function). This option provides the advantages of (i) speeding-up the process (for avoiding operations with files) and (ii) allowing the introduction of new features. Meanwhile, another alternative approach to overcome this limitation of the toolkit was also found. By defining a fixed name for the hydraulics file, the generation of temporary files is avoided and the information is always saved in the same file, which allows an unlimited number of iterations. A name for the hydraulics file can be defined in the initial input file of the network model (<network>.inp) by inserting a new line in the section [OPTIONS]: "Hydraulics SAVE <hydfilename>".

In the two procedures presented in Algorithm 7 for the energy costs computation, it can be observed that the main differences are related with the functions that execute the hydraulic simulation of the network ($SimulNoSaveHyd()$ or $SimulSaveHyd()$) and the functions to compute or obtain the computed costs ($ComputeMaxDemandCharge()$ and $ComputePumpCost()$ or $ReadRptCostand-Warnings()$, respectively). The last steps presented in both procedures implement the tanks and pressure penalty functions evaluation using exactly the same methods ($tankConstraints()$ and $pressure-Constraints()$).

Finally, the objective function is computed by adding the penalty functions to the pumping energy costs associated to the operation of the network during the simulated period (computed or obtained from EPANET). In fact, this value of the objective function already corresponds to the penalised objective function as discussed in the previous part of this thesis (Part III).

In each performed hydraulic simulation, a report file is generated by EPANET[§]. In some situations, warning and error messages generated during the simulation are written to such report. When running an EPANET hydraulic analysis, warning messages can occur by several reasons, such as (Rossman, 2000): (i) the system cannot converge to a stable/balanced hydraulic solution in the allowed number of trials/iterations (defined by the user in the input file), (ii) pumps and/or valves may

[‡]In Microsoft Windows® environment, it is only possible to create 65535 temporary files in a single directory. The .NET Framework mentions this limit related with the GetTempFileName method (Microsoft: developer network, 2015)

[§]Note that in the input file, <network>.inp, in section [REPORT], the option for energy computation should be added before running the optimisation by inserting the line "Energy yes".

Algorithm 7 Main steps for computing the pumping energy costs and/or for obtaining the value computed by EPANET.

function $OFcompute()$
    Execute a simulation without saving hydraulic results: $SimulNoSaveHyd()$;
    Compute costs associated to pumps operation at the peak power demand charge: $ComputeMaxDemandCharge()$;
    Compute pumping energy costs considering the power consumed and the operating time (Equation 8.3): $ComputePumpCost()$;
    Search for warning messages and define the warning-related constraint function (Equation 8.11): $ReadRptWarnings()$;
    Evaluate the water levels in tanks and define the tanks-related constraint functions (Equations 8.7 to 8.9): $TankConstraints()$;
    Evaluate the nodal pressures and define the pressure-related constraint function (Equation 8.10): $PressureConstraint()$;
    Compute the penalised objective function: $OF = C_{\text{total}} + F^{\text{H}}_{\text{warn}} + F^{\text{G}}_{\text{tanks}} + F^{\text{G}}_{\text{press}}$;
return OF.
end function

function OFEPANET()
    Execute a simulation saving the hydraulic results: $SimulSaveHyd()$;
    Read report file to obtain EPANET computed costs and search for warnings $ReadRptCostAndWarnings()$;
    Evaluate the water levels in tanks and define the tanks-related constraint functions (Equations 8.7 to 8.9): $TankConstraints()$;
    Evaluate the nodal pressures and define the pressure-related constraint function (Equation 8.10): $PressureConstraint()$;
    Compute the objective function: $OF = C_{\text{total}} + F^{\text{H}}_{\text{warn}} + F^{\text{G}}_{\text{tanks}} + F^{\text{G}}_{\text{press}}$;
    return OF.
end function

not be able to deliver enough flow if operating outside their limits, (iii) solution may leads to negative pressures in nodes, etc. Any kind of solution that produces warning messages is considered an infeasible solution and should not be accepted. To discard such solutions, the occurrence of warnings is treated as an equality constraint penalty function ($F^{\mathrm{H}}_{\mathrm{warn}}$) that is computed by adding a warning penalty to the function value each time a warning message is identified (as shown in Algorithm 8, step 5):

$$F^{\mathrm{H}}_{\mathrm{warn}} = r_{\mathrm{h,warn}} \times n_{\mathrm{warn}}, \tag{11.1}$$

where n_{warn} is the number of warnings that occurred in the hydraulic analysis and $r_{\mathrm{h,warn}}$ is the warning penalty coefficient pre-defined by the user (in file DEF_general.txt).

Algorithm 8 Procedure to account for the costs computed by EPANET and warning messages.

1: **function** READRPTCOSTANDWARNINGS()
2: Open OptimNet.rpt file;
3: Read total cost and demand charge cost computed by EPANET;
4: Obtain EPANET computed pumping energy costs: $C_{\mathrm{pumping}} = C_{\mathrm{total}} - C_{\mathrm{demandCharge}}$;
5: **while** Searching for warning messages **do**
6: **if** warning found **then**
7: $n_{\mathrm{warn}} = n_{\mathrm{warn}} + 1$;
8: $F^{\mathrm{H}}_{\mathrm{warn}} = r_{\mathrm{h,warn}} n_{\mathrm{warn}}$;
9: **end if**
10: **end while**
11: Close OptimNet.rpt file;
12: **return** $F^{\mathrm{H}}_{\mathrm{warn}}$ and C_{pumping}.
13: **end function**

As mentioned before, the main difference in the hydraulic simulation execution for computing the costs or obtaining the value computed by EPANET is related with the decision of saving the hydraulic results in the report file. The procedure to compute the pumping energy costs includes a simulation without saving the hydraulic results (*SimulNoSaveHyd*()) and uses the results saved during the simulation for the external computation of the pumping costs (*ComputePumpCost*()) and the additional costs related to the pumps operation at maximum power (*ComputeMaxDemandCharge*()). On the other side, the procedure to obtain the pumping costs computed by EPANET includes a simulation considering the hydraulic results (*SimulSaveHyd*()) and reads the computed values of pumping costs and additional costs from the report file (*ReadRptCostandWarnings*()). Algorithm 9 shows the steps implemented to perform the simulation with the option of saving the hydraulic results. The main difference occurs in step 3, where the $flag = 01$ or 00 defines if the initialised hydraulic analysis will be performed saving or not the results to a file. When the option of saving results is used, two additional functions are used to generate the results file: *ENsaveH*() and *ENreport*(), for saving and writing the hydraulic results in a report file (steps 30 and 31 of Algorithm 9). During the hydraulic analysis, the procedures are similar. For each *clockTime*, the power consumed by each pump as well as the delivered flow rate (already adjusted by the corresponding speed) are obtained by using the toolkit function *ENgetlinkvalue*(). The *clockTime* represents any hour with associated occurrences, such as a pipe or valve status change, which may not be coincident with the pre-defined hydraulic steps. Such hours with occurrences are also saved. For each pre-defined hydraulic step, the values of tanks water levels and nodal pressures are saved, respectively, for all tanks and junctions using the toolkit

function $ENgetnodevalue()$.

Algorithm 9 Network simulation and result saving.

```
1: procedure SIMULSAVEHYD()
2:     Open hydraulic solver: ENopenH();
3:     Define flag = 01 (save results to file);
4:     Initialise hydraulic analysis: ENinitH(flag);
5:     while time period > 0 do
6:         Run hydraulic analysis: ENrunH(clockTime);
7:         for each clockTime do
8:             Save clockTime;
9:             for all pumps do
10:                Save the obtained pump power (P_EPA): ENgetlinkvalue();
11:                Save the obtained pump flow rate (Q_2,EPA): ENgetlinkvalue();
12:            end for
13:            if clockTime equal to pre-defined step then
14:                for all junctions do
15:                    Save the obtained nodal pressure: ENgetnodevalue();
16:                end for
17:                for all tanks do
18:                    Save the obtained current water level (pressure): ENgetnodevalue();
19:                end for
20:            end if
21:        end for
22:        Calculate the length of the next time period: ENnextH(time period);
23:    end while
24:    switch efficiencyDecision do
25:        case efficiencyDecision =EPA
26:            Maintain the same values saved for the pumps power;
27:        case efficiencyDecision =AL or SB or CAC
28:            Compute the values of power considering the selected efficiency formula:
               changeEfficiency();
29:    Close hydraulic solver: ENcloseH();
30:    Save hydraulic results: ENsaveH();
31:    Write results in the report file: ENreport();
32:    Close toolkit: ENclose();
33: end procedure
```

After performing the hydraulic analysis and obtaining the values of power consumed by each pump in each step, a correction of such values taking into account the formulation for the pumps efficiency computation may be executed according to the *efficiencyDecision*. In case the user intends to compute the values of pump power consumption considering a specific method for the pumps efficiency computation with speed variation (see Section 7.2), the switch *efficiencyDecision* should be set to (i) AL, for considering the Affinity Laws, (ii) SB for considering the Sarbu and Borza method or (iii) CAC for considering the new method proposed in this work. To keep the same values of power consumption computed by EPANET, the switch should be set to EPA. The procedure implemented for the pump power correction is presented in Algorithm 10.

Algorithm 10 Steps for the pumps power correction.

procedure CHANGEEFFICIENCY()
  for all variable-speed pumps **do**
    Obtain the EPANET pump efficiency ($\eta_{2,\text{EPA}}$) for all periods of time from interpolation in the efficiency curve using the saved pump flow values ($Q_{2,\text{EPA}}$);
    Determine the values of pump flow rate at nominal speed: $Q_1 = Q_{2,\text{EPA}}/(N_2/N_1)$
    Obtain the efficiency values (η_1) for each corresponding flow rate, Q_1, by interpolation in the pump efficiency curve;
  end for
  if $efficiencyDecision$ =AL **then**
    Set $\eta_{2,\text{AL}} = \eta_1$;
    Compute the corrected power values according to the Affinity Laws: $P_{\text{AL}} = P_{\text{EPA}} \left(\frac{\eta_{2,\text{EPA}}}{\eta_{2,\text{AL}}} \right)$;
  else if $efficiencyDecision$ =SB **then**
    Compute the speed-adjusted efficiency values according to the Sarbu and Borza (SB) formula: $\eta_{2,\text{SB}} = 1 - (1 - \eta_1) \left(\frac{N_1}{N_2} \right)^{0.1}$;
    Compute the corrected power values according to the SB method: $P_{\text{SB}} = P_{\text{EPA}} \left(\frac{\eta_{2,\text{EPA}}}{\eta_{2,\text{SB}}} \right)$;
  else if $efficiencyDecision$ =CAC **then**
    Compute the speed-adjusted efficiency values according to the formula proposed in this work (CAC): $\eta_{2,\text{CAC}} = \eta_1 \left[\left(\frac{N_2}{N_1} - 1 \right)^3 + 1 \right]$;
    Compute the corrected power values according to the proposed method: $P_{\text{CAC}} = P_{\text{EPA}} \left(\frac{\eta_{2,\text{EPA}}}{\eta_{2,\text{CAC}}} \right)$;
  end if
end procedure

The first steps consist in determine the efficiency value computed by EPANET for each variable-speed pump in all periods of time (considering the *clockTime* values). From the efficiency curve used by EPANET for each specific pump and using the delivered flow rate in each interval, the EPANET pump efficiency, $\eta_{2,\text{EPA}}$, can be obtained by interpolation of the flow values in the corresponding efficiency curve. After that, the pump flow rate at nominal speed, Q_1, is computed using the affinity law for flow (step 4 of Algorithm 10). Using the same efficiency curve always used by EPANET, the values of pump efficiency at nominal speed, η_1, can also be obtained by interpolation (see Figure 11.2).

Since the pump power computed by EPANET is given by:

$$P_{EPA} = \frac{\gamma Q_2 H_2}{\eta_{2,\text{EPA}}}, \tag{11.2}$$

then, the method implemented to correct the values of pump power multiplies the obtained values by the EPANET computed efficiency, $\eta_{2,\text{EPA}}$, and divides it by the new speed-adjusted computed efficiency, $\eta_{2,\text{corrected}}$ (*i.e.* $\eta_{2,\text{AL}}$, $\eta_{2,\text{SB}}$ or $\eta_{2,\text{CAC}}$):

$$P_{\text{corrected}} = P_{\text{EPA}} \frac{\eta_{2,\text{EPA}}}{\eta_{2,\text{corrected}}}. \tag{11.3}$$

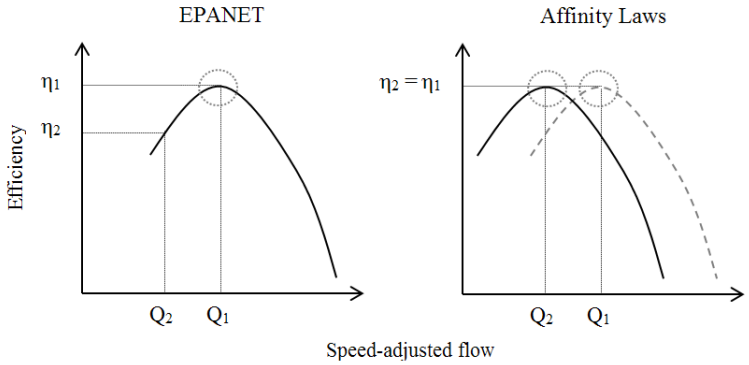


Figure 11.2: Scheme to demonstrate the calculation of the pump efficiency based on the Affinity Laws from the efficiency computed by EPANET.

In the particular case of the efficiency computation considering the affinity laws, since it is only expected a deviation of the nominal efficiency curve with the speed change, this means that the value of pump efficiency is maintained even considering the speed-adjusted flow rate (see demonstration in Figure 11.2). Thus, the correction of the speed-adjusted efficiency computed by EPANET to the efficiency computed by the AL ($\eta_{2,\text{AL}}$) can be performed by considering that $\eta_{2,\text{AL}}$ is equal to the pump efficiency at nominal speed (η_1).

For the other two possible methods for the efficiency computation, the corresponding formulae should be applied, according to step 11 or step 14 of Algorithm 10 and replaced in Equation 11.3.

After executing the hydraulic simulation of the network operation to obtain the power consumption, the pumping energy can finally be obtained. Considering the process of obtaining the costs computed by EPANET (function $OFepanet()$ in Algorithm 7), only the function $ReadRptCostand-Warnings()$ is called. This function, whose implementation steps are presented in Algorithm 8, opens

the report file generated after the hydraulic simulation[¶] and reads the contained information of the operational costs, namely the total costs and the demand charge. Although these additional costs are not typically considered in studies involving the evaluation of operating costs, it is included in the total costs computed by EPANET. Thus, the pumping energy costs were computed by subtracting the $C_{\text{demandCharge}}$ to the C_{total} (step 4 in Algorithm 8). Considering the process for computing the pumping costs (function $OFcompute()$ in Algorithm 7), in order to obtain the same results as the computed by EPANET, two functions were implemented to compute the pumping energy costs ($ComputePumpCost()$) and the additional demand charge costs ($ComputeMaxDemandCharge()$). The main steps implemented in each function are, respectively, listed in Algorithms 11 and 12.

Algorithm 11 Computation of pumping energy costs.

1: **function** COMPUTEPUMPCOST()
2: **for** All pumps **do**
3: Compute the cost of energy consumed in each time period of operation: $C_{\text{pumping},p,s} = P_{p,s} t_{\text{op},p,s} T_{p,s}$, where P corresponds to the values of power saved and $T_{p,s}$ is the value of the tariff associated to each pump p in the time period s;
4: **end for**
5: Sum the costs computed for all pumps in all time periods: $C_{\text{pumping}} = \sum_p \sum_s C_{\text{pumping}p,s}$;
6: **return** C_{pumping}.
7: **end function**

The pumping energy costs are computed using the power consumed in each time interval and the corresponding price of energy in such interval of time. The total pumping energy costs associated to the operation of the network is the sum of the energy costs associated to each pump in each time interval.

Algorithm 12 Computation of the additional cost associated to pumps operations at peak demand.

1: **function** COMPUTEMAXDEMANDCHARGE()
2: Find peak power, $P_{\max}$ from all the values of power saved (P_{EPA});
3: Sum all values of power equal to the peak (Equation 8.4) $\alpha_p = P_{\max} n_{P_{\max}}$;
4: Compute the additional costs due to pumps operation at maximum power (Equation 8.3): $C_{\text{demandCharge}} = \sum_{p=1}^{n_{\text{pumps}}} \text{DC}_p \alpha_p$, where DC is the demand charge pre-defined by the user in the <network>.inp file;
5: **return** $C_{\text{demandCharge}}$.
6: **end function**

The function for the computation of the additional costs uses the same values of power consumption, determines the peak/maximum power of such set of values and sums all values of power that equals the peak, *i.e.*, multiplies the peak power by the number of times each pump is operating at the same maximum power (step 3 in Algorithm 12). Finally, the result is multiplied by the demand charge (DC) pre-defined by the user in order to obtain the same $C_{\text{demandCharge}}$ as the computed by EPANET.

To obtain the value of the augmented objective function, the penalty functions should be computed and added to the value of pumping energy costs (such as previously shown in steps 8 and 16 of Algorithm 7). The steps implemented for computing the penalty functions in case of violation of both

[¶]Remember that the generated files can only be used for read/write after closing the toolkit, explaining why the $ENclose()$ function is used in the end of the simulation procedure.

tanks-related and pressure-related constraints are provided in Algorithms 13 and 14, respectively.

As shown in Algorithm 13, the first constraints checked are the related with the minimum and maximum alarm levels of each tank of the network. Instead of comparing the water level in tanks in each hour with the limit operating levels (minimum and maximum allowed levels of operation) defined in the input file for each tank, the measured water levels in each hour of the simulation are compared with the alarm levels, which correspond to more restrict limits above the maximum and below the minimum. This was implemented because in some water utilities there are in fact two distinct types of limits considered for the tanks operation: the operating limit and the alarm limit. The last one cannot, in any situation, be overcome. Thus, the user has the option of providing the values that want to consider for both minimum and maximum levels. The difference between the operating and alarm levels for both the minimum and maximum levels of the tanks can be pre-defined by the user in terms of percentage. In case the user only wants to consider the minimum and maximum levels defined in the model of the network, the percentage for both the minimum level difference ($L_{\%\text{minSecurity}}$) and the maximum level difference ($L_{\%\text{maxSecurity}}$) should be set to zero. Otherwise, the user should introduce the tanks alarm levels in the input file of the network model (<network>.inp) and insert the desired percentage for the difference between the alarm level and the operating level. It should be noticed that the use of operating levels for the constraints evaluation ensures that the resulting solutions from the optimiser will never overcome the alarm levels. According to the implemented steps 3 to 32 presented in Algorithm 13, the maximum level penalty function ($F^{\text{G}}_{\text{maxLevel}}$) ensures that the water levels never exceed the maximum operating levels. At the same time, the minimum level penalty function ($F^{\text{G}}_{\text{minLevel}}$) ensures the water levels never drop below the minimum operating levels.

The other type of constraint related with the tanks depends on a user pre-defined option and concerns the water levels in the end of the simulation period. The user has four available options: (i) consider continuity, *i.e.* the water level in the end of the simulation should be equal to the water level in the beginning, (ii) do not consider continuity, meaning that no constraint will be considered, (iii) final level superior or equal to the initial (inequality constraint), allowing to accept more solutions then considering the continuity constraint, or (iv) consider a pre-defined final water level, which limits the optimisation process similarly with the continuity constraint. An appropriate penalty method is implemented according to the type of constraint (equality or inequality) such as the discussed in the previous part of the thesis.

Finally, the tanks-related penalty function is obtained by summing the three distinct computed penalties, *i.e.*

$$F^{\text{G}}_{\text{tanks}} = F^{\text{G}}_{\text{maxLevel}} + F^{\text{G}}_{\text{minLevel}} + F^{\text{G}}_{\text{finalLevel}}. \tag{11.4}$$

Similar with the minimum level penalty function, the pressure-related penalty function also applies a penalty in case the values of pressure in each junction drop below the minimum value of pressure pre-defined by the user.

The dimension of each implemented penalty function is dependent on the penalty coefficients defined by the user for each case. To ignore a certain constraint, the corresponding penalty coefficient should always be set to zero. This is valid for all the coefficients: $r_{\text{g,maxLevel}}$, $r_{\text{g,minLevel}}$, $r_{\text{g,finalLevel}}$, $r_{\text{g,minPress}}$ and $r_{\text{h,warn}}$.

It should be noticed that the discussed implemented methodologies concerning the simulation options, as mentioned before, can be used separately from the optimisation module in case the user

Algorithm 13 Computation of the tanks-related penalty functions.

function TANKSCONSTRAINTS()
  Initialise variables for the constraint functions values: $h_{1,i} = 0$, $g_{1,i} = 0$, $g_{2,i} = 0$, $g_{3,i} = 0$;
  for Each tank i **do**
    Define the maximum operating level: $L_{i,\max} = L_{i,\text{maxAlarm}} - (L_{\%\text{maxSecurity}} \times L_{i,\text{maxAlarm}})$;
    Define the minimum operating level: $L_{i,\min} = L_{i,\text{minAlarm}} - (L_{\%\text{minSecurity}} \times L_{i,\text{minAlarm}})$;
    for Each time step **do**
      if $L_i > L_{i,\max}$ **then**
        $g_{2,i} = L_i - L_{i,\max}$;
        $F^{\text{G}}_{\text{maxLevel}} = F^{\text{G}}_{\text{maxLevel}} + r_{\text{g,maxLevel}} [\max\{0, g_{2,i}\}]^2$;
      end if
      if $L_i < L_{i,\min}$ **then**
        $g_{3,i} = L_{i,\min} - L_i$;
        $F^{\text{G}}_{\text{minLevel}} = F^{\text{G}}_{\text{minLevel}} + r_{\text{g,minLevel}} [\max\{0, g_{3,i}\}]^2$;
      end if
    end for
    switch *finalLevelDecision* **do**
      case *finalLevelDecision* = continuity
        if $L_{i,\text{final}} \neq L_{i,\text{inital}}$ **then**
          $h_{1,i} = L_{i,\text{final}} - L_{i,\text{inital}}$;
          $F^{\text{H/G}}_{\text{finalLevel}} = F^{\text{H/G}}_{\text{finalLevel}} + r_{\text{h,finalLevel}} [h_{1,i}]^2$;
        end if
      case *finalLevelDecision* = noContinuity
        $F^{\text{H/G}}_{\text{finalLevel}} = 0$;
      case *finalLevelDecision* = superiorOREqual
        if $L_{i,\text{final}} < L_{i,\text{inital}}$ **then**
          $g_{1,i} = |L_{i,\text{final}} - L_{i,\text{inital}}|$;
          $F^{\text{H/G}}_{\text{finalLevel}} = F^{\text{H/G}}_{\text{finalLevel}} + r_{\text{g,finalLevel}} [\max\{0, g_{1,i}\}]^2$;
        end if
      case *finalLevelDecision* = definedLevel
        if $L_{i,\text{final}} \neq L_{i,\text{defined}}$ **then**
          $h_{1,i} = L_{i,\text{final}} - L_{i,\text{defined}}$;
          $F^{\text{H/G}}_{\text{finalLevel}} = F^{\text{H/G}}_{\text{finalLevel}} + r_{\text{h,finalLevel}} [h_{1,i}]^2$;
        end if
  end for
  Compute the tanks-related penalty function: $F^{\text{H/G}}_{\text{tanks}} = F^{\text{G}}_{\text{minLevel}} + F^{\text{G}}_{\text{maxLevel}} + F^{\text{H/G}}_{\text{finalLevel}}$;
  return $F^{\text{H/G}}_{\text{tanks}}$.
end function

Algorithm 14 Computation of the pressure-related penalty function.

function PRESSURECONSTRAINT()
  Initialise the variable for the penalty value: $g_{4,i} = 0$;
  for Each junction i **do**
    if $P_i < P_{i,\min}$ **then**
      $g_{4,i} = |P_i - P_{i,\min}|$;
      $F^{\mathrm{G}}_{\mathrm{minPress}} = F^{\mathrm{G}}_{\mathrm{minPress}} + r_{\mathrm{g,minPress}} \times [\max\{0, g_{4,i}\}]^2$;
    end if
  end for
  return $F^{\mathrm{G}}_{\mathrm{minPress}}$.
end function

only wants to perform hydraulic analysis to the modelled network. In this case, the same procedures described in Algorithm 6 for the evaluation of the objective function, which in fact corresponds to the simulation of the network operation, are performed excluding the steps concerning the variables update.

11.2 Control optimisation module

The optimisation approach followed in this thesis considers the control of both pumps and valves of the water supply and distribution networks. Two types of decision variables are then considered to improve the network control with the aim of reducing the operating costs: (i) time rates, t_{op}, *i.e.* the time of pumps operation in each time-step and the opening time of valves in the same step, and (ii) relative speeds of the variable-speed pumps in each operating interval (time-step), M. The method implemented to control the variables set, $\mathbf{X}$, in an EPANET model was based in the use of both pump speed patterns and time controls. A speed pattern containing the pump speed setting (relative speed) of each time-step of the simulation is associated to each variable-speed pump. Concerning the time controls, each element/component (pump and/or valve), whose operational control is intended to be found in order to minimise costs, must have associated one control statement for each time-step of the simulation indicating the status (or speed setting). This means that a pump operating during a day divided into 24 time-steps must have associated 24 time controls. Table 11.1 provides examples of controls associated to the three possible elements to control: a variable-speed pump, a fixed-speed pump and a valve. The variable-speed pump is operating at 70 % of its nominal speed (0.7) in the first step and then, when operating, is always at nominal speed (1.0). Both the fixed-speed pump and the valve are open during all day except in the last two hours of simulation (from 22h to 24h).

When the user sets the option to optimise the network operation, the optimisation process is automatically initialised according to the procedures presented in Algorithm 15. The first procedure performs the initialisation of the objective function ($initialiseOF()$), which mainly involves the preparation of the decision variables. Using the information of the network initially saved in memory, a vector containing the values of the decision variables (speed and time rates) is defined. Depending on the user pre-defined option for the switch *aggregDecision*, the values for all time-steps can be directly introduced in the vector of decision variables through the developed function $defineX()$, or, alternatively, a search for possible aggregations of time-steps in blocks can be performed (if $aggregDecision = true$) using the developed function $aggregateX()$ (see Section 11.2.1). The ini-

Table 11.1: Examples of time control statements implemented in the network model for the optimisation of a variable-speed pump (pump1), a fixed-speed pump (pump2) and a Valve.

Variable-speed pump	Fixed-speed pump	Valve
LINK Pump1 0.7000 AT TIME 0.0000	LINK Pump2 OPEN AT TIME 0.0000	LINK Valve OPEN AT TIME 0.0000
LINK Pump1 1.0000 AT TIME 1.0000	LINK Pump2 OPEN AT TIME 1.0000	LINK Valve OPEN AT TIME 1.0000
LINK Pump1 0.0000 AT TIME 2.0000	LINK Pump2 OPEN AT TIME 2.0000	LINK Valve OPEN AT TIME 2.0000
LINK Pump1 1.0000 AT TIME 3.0000	LINK Pump2 OPEN AT TIME 3.0000	LINK Valve OPEN AT TIME 3.0000
LINK Pump1 1.0000 AT TIME 4.0000	LINK Pump2 OPEN AT TIME 4.0000	LINK Valve OPEN AT TIME 4.0000
LINK Pump1 1.0000 AT TIME 5.0000	LINK Pump2 OPEN AT TIME 5.0000	LINK Valve OPEN AT TIME 5.0000
LINK Pump1 1.0000 AT TIME 6.0000	LINK Pump2 OPEN AT TIME 6.0000	LINK Valve OPEN AT TIME 6.0000
LINK Pump1 1.0000 AT TIME 7.0000	LINK Pump2 OPEN AT TIME 7.0000	LINK Valve OPEN AT TIME 7.0000
LINK Pump1 1.0000 AT TIME 8.0000	LINK Pump2 OPEN AT TIME 8.0000	LINK Valve OPEN AT TIME 8.0000
LINK Pump1 1.0000 AT TIME 9.0000	LINK Pump2 OPEN AT TIME 9.0000	LINK Valve OPEN AT TIME 9.0000
LINK Pump1 1.0000 AT TIME 10.0000	LINK Pump2 OPEN AT TIME 10.0000	LINK Valve OPEN AT TIME 10.0000
LINK Pump1 1.0000 AT TIME 11.0000	LINK Pump2 OPEN AT TIME 11.0000	LINK Valve OPEN AT TIME 11.0000
LINK Pump1 1.0000 AT TIME 12.0000	LINK Pump2 OPEN AT TIME 12.0000	LINK Valve OPEN AT TIME 12.0000
LINK Pump1 1.0000 AT TIME 13.0000	LINK Pump2 OPEN AT TIME 13.0000	LINK Valve OPEN AT TIME 13.0000
LINK Pump1 1.0000 AT TIME 14.0000	LINK Pump2 OPEN AT TIME 14.0000	LINK Valve OPEN AT TIME 14.0000
LINK Pump1 1.0000 AT TIME 15.0000	LINK Pump2 OPEN AT TIME 15.0000	LINK Valve OPEN AT TIME 15.0000
LINK Pump1 1.0000 AT TIME 16.0000	LINK Pump2 OPEN AT TIME 16.0000	LINK Valve OPEN AT TIME 16.0000
LINK Pump1 1.0000 AT TIME 17.0000	LINK Pump2 OPEN AT TIME 17.0000	LINK Valve OPEN AT TIME 17.0000
LINK Pump1 1.0000 AT TIME 18.0000	LINK Pump2 OPEN AT TIME 18.0000	LINK Valve OPEN AT TIME 18.0000
LINK Pump1 1.0000 AT TIME 19.0000	LINK Pump2 OPEN AT TIME 19.0000	LINK Valve OPEN AT TIME 19.0000
LINK Pump1 1.0000 AT TIME 20.0000	LINK Pump2 OPEN AT TIME 20.0000	LINK Valve OPEN AT TIME 20.0000
LINK Pump1 1.0000 AT TIME 21.0000	LINK Pump2 OPEN AT TIME 21.0000	LINK Valve OPEN AT TIME 21.0000
LINK Pump1 0.0000 AT TIME 22.0000	LINK Pump2 CLOSED AT TIME 22.0000	LINK Valve CLOSED AT TIME 22.0000
LINK Pump1 0.0000 AT TIME 23.0000	LINK Pump2 CLOSED AT TIME 23.0000	LINK Valve OPEN AT TIME 23.0000

tial vector generated will be used as an initial solution of the first optimisation algorithm called.

After the preparation of the decision variables, the optimisation process is started ($startOptimisation()$) using a sequential list of algorithms pre-defined by the user in the input file of the general definitions.

The following five algorithms were implemented and can be selected by the user:

1. Nelder-Mead Simplex (NMSimplex) - the source code of the Nelder-Mead simplex method for unconstrained optimisation, developed by Michael F. Hutt (Hutt, 2007), was implemented and adapted for a N-dimensional space.

2. Genetic Algorithms (GA) - the source code of a simple real coded genetic algorithm, developed by Takahama (Takahama, 2005c), was implemented and adapted for a N-dimensional space.

3. Differential Evolution (DE) - the source code of the simplest variant of DE algorithm, developed by Takahama (Takahama, 2005a), was implemented and adapted.

4. Particle Swarm Optimisation (PSO) - the source code of a simple PSO algorithm, developed by Takahama (Takahama, 2005b), was also implemented and adapted.

5. Adaptive Simulated Annealing (ASA) - a variant of simulated annealing (SA), developed by Lester Ingber in C-language code (Ingber, 2015), was implemented. In this variant of SA, the algorithm parameters are automatically adjusted (adapted) according to the optimisation progress.

All the implemented optimisation algorithms handle constraints through static exterior penalty methods, as previously described in section 11.1. The user only needs to introduce the desired penalty

Algorithm 15 General optimisation procedure.

```
1: procedure INITIALISEOF()
2:     switch aggregDecision do
3:         case true
4:             Define reduced matrix of decision variables X: aggregateX();
5:         case false
6:             Define matrix of decision variables X: defineX();
7: end procedure
8: procedure STARTOPTIMISATION()
9:     for Each S in sequence list do
10:        if algorithm in position S = NMSimplex then
11:            Run the NMSimplex algorithm to find the optimal solution:
               X_opt = NMSimplexSolution();
12:            Replace the vector of decision variables by the new best found: X = X_opt;
13:        end if
14:        if algorithm in position S = GA then
15:            Run the GA algorithm to find the optimal solution: X_opt = GASolution();
16:            Replace the vector of decision variables by the new best found: X = X_opt;
17:        end if
18:        if algorithm in position S = DE then
19:            Run the DE algorithm to find the optimal solution: X_opt = DESolution();
20:            Replace the vector of decision variables by the new best found: X = X_opt;
21:        end if
22:        if algorithm in position S = PSO then
23:            Run the PSO algorithm to find the optimal solution: X_opt = PSOSolution();
24:            Replace the vector of decision variables by the new best found: X = X_opt;
25:        end if
26:        if algorithm in position S = ASA then
27:            Run the ASA algorithm to find the optimal solution: X_opt = ASASolution();
28:            Replace the vector of decision variables by the new best found: X = X_opt;
29:        end if
30:    end for
31: end procedure
```

coefficients for each distinct type of constraint. However, in case of not considering a particular constraint, its coefficient should be set to zero.

There are two possible types of optimisation techniques: (i) optimisation with a single algorithm or (ii) sequential optimisation with two or more algorithms.

As can be observed in the procedure represented in Algorithm 15, for each sequence list, the search by the optimal solution $\mathbf{X}_{\text{opt}}$ is performed by using the selected algorithm for that position in the sequence, *i.e.* by calling the adapted functions of each optimisation algorithm (*NMSimpleSolution()*, *GASolution()*, *DESolution()*, *PSOSolution()* and *ASASolution()*). The first optimisation algorithm considers the initial solution obtained from the network input file and the following algorithms use the best solution obtained with the previous one. However, population-based algorithms also use random sets as initial solutions.

11.2.1 Definition of the decision variables matrix and the aggregation technique

In the process of defining the matrix of the decision variables, a technique for grouping/aggregating the variables into larger time-steps is available. This aggregation technique may allow reducing the number of variables as well as the number of pumps shut-off with a minimum loss in the universe of solutions.

The pseudo-codes of the steps implemented to define the matrix of decision variables without and with blocks aggregation are described in Algorithms 16 ($defineX()$ procedure) and 17 ($aggregateX()$ procedure), respectively.

Algorithm 16 Definition of the matrix of decision variables.

procedure DEFINEX()
  Compute the number of decision variables: $n_{\text{var}} = n_{steps}\left(2n_{\text{VSP,opt}} + n_{\text{FSP,opt}} + n_{\text{Valves,opt}}\right)$;
  Define the matrix of decision variables $\mathbf{X} = (x_{i,j}) \in \mathbb{R}^{n_{\text{comp}} \times n_{\text{steps}}}$;
  Initialise i and j;
  for All pumps to optimise ($n_{\text{VSP,opt}} + n_{\text{FSP,opt}}$) **do**
    for Each j time-step **do**
      Normalise the operating time as *pumpTimeRate* and save in the matrix of decision variables (Equation 11.5): $x_{i,j} = t_{\text{op}}/t_{\text{step}}$;
    end for
    if pump is variable-speed type **then**
      for Each j time-step **do**
        Normalise the pump speed and save in the matrix of decision variables (Equation 11.6): $x_{i,j} = \frac{M - M_{\text{min}}}{M_{\text{max}} - M_{\text{min}}}$, where M_{min} and M_{max} are the user defined speed limits;
      end for
    end if
  end for
  for All $n_{\text{Valves,opt}}$ valves to optimise **do**
    for Each j time-step **do**
      Normalise the valve opening time as *valveTimeRate* and save in the matrix of decision variables (Equation 11.5): $x_{i,j} = t_{\text{op}}/t_{\text{step}}$;
    end for
  end for
end procedure

Algorithm 17 Definition of the reduced matrix of decision variables using the aggregation technique.

procedure AGGREGATEX()
  Compute the initial number of decision variables: $n_{\text{var}} = n_{steps}\left(2n_{\text{VSP,opt}} + n_{\text{FSP,opt}} + n_{\text{Valves,opt}}\right)$;
  Search for aggregated blocks and definition of the matrix of aggregated steps: *buildAggregStep*();
  Build the correspondence matrix between the number of decision variables, n_{var}, and n_{aggreg}: *buildCorrespVector*();
  Define current number of decision variables according to the obtained **AggregStep** matrix: *computeNaggreg*();
  Define the matrix of initial decision variables: $\mathbf{X} = (x_{i,j})$ with $x_{i,j} \in [0,1]$;
  Define i and j in the matrix of the initial decision variables $x_{i,j}$;
  Define the position k in the correspondence matrix $\mathbf{correspVector}_k$;
  for All pumps to optimise $(n_{\text{VSP,opt}} + n_{\text{FSP,opt}})$ **do**
    for Each j time-step **do**
      if $\mathbf{correspVector}_k > 0$ **then**
        Normalise the operating time as pumpTimeRate and save in the matrix of decision variables (Equation 11.5): $x_{i,j} = t_{\text{op}}/t_{\text{step}}$;
      end if
      Go to next k;
    end for
    if pump is variable-speed type **then**
      for Each j time-step **do**
        if $\mathbf{correspVector}_k > 0$ **then**
          Normalise the pump speed and save in the matrix of decision variables (Equation 11.6): $x_{i,j} = \frac{M - M_{\text{min}}}{M_{\text{max}} - M_{\text{min}}}$, where M_{min} and M_{max} are the user defined speed limits;
        end if
        Go to next k;
      end for
    end if
  end for
  for All valves to optimise $n_{\text{Valves,opt}}$ **do**
    for All steps **do**
      if $\mathbf{correspVector}_k > 0$ **then**
        Normalise the valve opening time as valveTimeRate and save in the matrix of decision variables (Equation 11.5): $x_{i,j} = t_{\text{op}}/t_{\text{step}}$;
      end if
      Go to next k;
    end for
  end for
end procedure

When the *aggregDecision* switch is set to *false*, the procedure starts by determining the number of decision variables. This depends on the user decision concerning the elements that are intended to optimise: number variable-speed pumps ($n_{\text{VSP,opt}}$), number of fixed-speed pumps ($n_{\text{FSP,opt}}$) and number of valves ($n_{\text{Valves,opt}}$). Thus, for each one of these elements, the corresponding values of time rate (and speed, for variable-speed pumps) for all time-steps of the simulation are scaled/normalised between 0 and 1 (according to equation 8.1) and saved into the matrix of decision variables by the order each element appears in the network input file (<network>.inp), first for the pumps and then for the valves. The value of the variables that correspond to the normalised pump/valve time rate are obtained by dividing its operating/opening time by the dimension of the time-step (step 7 in Algorithm 16):

$$x_{i,j} = pump/valveTimeRate = \frac{\text{operating/opening time}}{\text{step size}} = \frac{t_{\text{op}}}{t_{\text{step}}} \tag{11.5}$$

where i and j are, respectively, the i^{th} component (pump operating time or valve opening time) and the j^{th} time-step in the matrix of the decision variables.

For a pump operating at variable-speed, its speed values for all steps are also normalised and saved after the values of time rate. This normalisation takes into account the pre-defined values for the minimum and maximum speeds allowed for the pumps (M_{min} and M_{max}) and is computed by (see step 11 in Algorithm 16):

$$x_{i,j} = \frac{M - M_{\text{min}}}{M_{\text{max}} - M_{\text{min}}}, \tag{11.6}$$

where N is the speed of the pump in the current j time-step.

The main difference of the $aggregateX()$ procedure to the $defineX()$ procedure is the additional steps for verification of possible aggregation of time-steps and the definition of the (reduced) number of decision variables taking into account the number of aggregated blocks[‖]. The search for possible aggregated blocks (step 3 in Algorithm 17) is performed by calling the function $builAggregStep()$, whose implemented steps are described in Algorithm 18. The *buildAggregStep()* function returns a vector containing the information about the aggregated blocks that replace the several variables of the successive time-steps by the variables representative of a single step. Figure 11.3 provides an example for a better explanation of this implemented aggregation process.

The process of searching for possible aggregation of variables starts by an analysis to all water demand patterns associated to the consumption nodes and all energy price patterns (tariffs) associated to the pumps. The idea is to find consecutive periods of time (time-steps) whose values (of demand and price) remain the same. Taking the example presented in Figure 11.3, observing the demand pattern it is possible to identify a block in the three first steps with the same value of demand, meaning that these three steps can be aggregated. In the case of the tariff, two blocks are identified, the first corresponding to the aggregation of 8 steps and the second of 16 steps. Notice that, if there was no demand pattern in this example, the initial 24 decision variables would be grouped into two blocks (the blocks identified in the energy price pattern), reducing the number of decision variables to 2 (or 4, in case of a variable-speed pump). However, in this case, the only blocks that should be considered are the ones that present constant consecutive values in the same steps for both the demand and the tariff. See, for instance, the first two blocks considered for each pattern. Besides the tariff presents

[‖] An aggregated block is a set of successive time-steps in which the network demands and the tariff remain constant. This set of time-steps is replaced by a single time-step whose size corresponds to the sum of the size of the replaced time-steps.

Algorithm 18 Analysis of aggregation blocks.

function BUILDAGGREDSTEP()
  Define matrix to save aggregation information, **AggregStep**;
  Analysis of water demands for steps aggregation:
  Define vector of demand aggregated steps, **aggregDemand**, of size n_{steps};
  for All s steps **do**
    Define vector to count steps, **stepCount**;
    for All nodes with water demand (or water inlet) **do**
      Define and initialise $boolValue = true$ and $aggregcount = 0$;
      Obtain demand pattern associated to current node;
      while $boolValue = true$ **do**
        if current demand differs from demand in step $(s+1+aggregCount)$ **then**
          Save value of $aggregCount$ in **stepCount**;
          Set $boolValue = false$;
        else$aggregCount = aggregCount + 1$;
        end if
      end while
    end for
    Found minimum value of **stepCount** and save in **aggregDemand**;
  end for
  Analysis of energy price patterns for steps aggregation:
  Define vector of tariff aggregated steps, **aggregTariff**;
  for All s steps **do**
    Define vector to count steps, **stepCount**;
    for All pumps **do**
      Define and initialise $boolValue = true$ and $aggregcount = 0$;;
      Obtain energy price (tariff) pattern associated to current pump;
      while $boolValue = true$ **do**
        if current tariff differs from tariff in step $(s+1+aggregCount)$ **then**
          Save value of $aggregCount$ in **stepCount**;
          Set $boolValue = false$;
        else $aggregCount = aggregCount + 1$;
        end if
      end while
    end for
    Found minimum value of **stepCount** and save in **aggregTariff**;
  end for
  Definition of a vector containing the possible steps aggregation information:
  for All s steps **do**
    Save number of common aggregated steps:
    $AggregStep_s = \min\{aggregDemand_s, aggregTariff_s\} + 1$;
    if $\min\{aggregDemand_s, aggregTariff_s\} > 0$ **then**
      Jump the following similar steps: $s = s + AggregStep_s - 1$;
    end if
  end for
return AggregStep;
end function

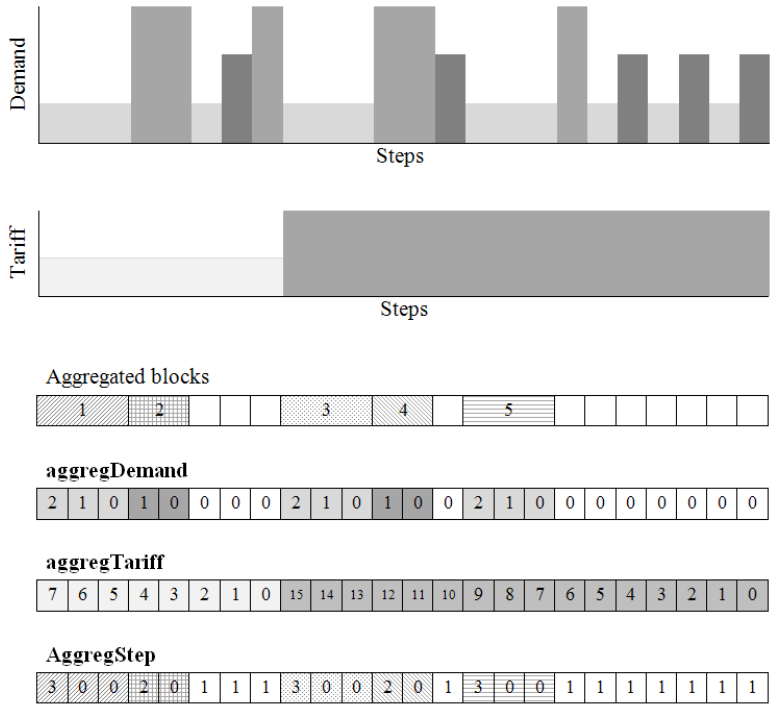


Figure 11.3: Demonstration of the *builAggregStep*() function using an example with one demand pattern and one energy price pattern (tariff).

a block with 8 steps, the demand only presents the first 3 steps constant, which means that the first possible aggregation block has a size of 3 steps. The second block of the demand pattern can be also considered an aggregation block since the corresponding steps for the tariff also remains constant. The same procedure is performed for the remaining steps. Thereby, the resulting number of possible blocks to aggregate is 5, meaning that the number of decision variables can be reduced from 24 to 16 (a reduction of 33 %).

The *buildAggregStep*() function essentially defines a vector (**aggregDemand**) that saves the information concerning the consecutive constant steps that are found in common between all the demand patterns (steps 3 to 19 in Algorithm 18). Another similar vector (**aggregTariff**) is also defined to keep the information related with the common blocks common for all the tariffs (steps 20 to 36 in Algorithm 18). After that, the two vectors are compared and a vector containing the sizes of the possible blocks to aggregate (**AggregStep**) can then be defined, such as the previously demonstrated through Figure 11.3.

Returning to the procedure *aggregateX*() (Algorithm 17), after obtaining the vector of the aggregated steps (**AggregStep**), a correspondence between the total number of variables (for all pumps and valves) and the aggregated ones should be defined and saved to a vector for a later use (to perform the disaggregation process after each optimisation iteration). The *buildCorrespVector*() function was implemented to perform such operation. The corresponding matrix is generated with the size of the initial number of variables where the values of the **AggregStep** are saved over the elements being optimised.

After obtaining the correspondence matrix (**correspMatrix**), the updated number of decision variables, considering the aggregated blocks is computed through the function *computeNaggreg*().

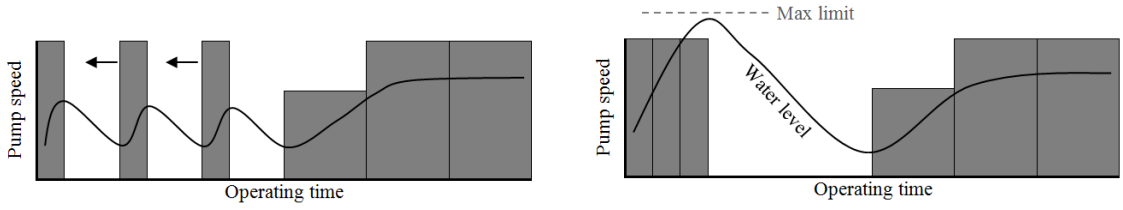
Finally, the process of defining the vector of decision variables in the *aggregateX()* procedure (Algorithm 17) is similar to the explained for the *defineX()* procedure considering only the steps where the $mathbf{corresp Matrix}$ values are non-zero.

After preparing the decision variables to send the initial solution to the optimiser, a procedure to send the optimal/new decision variables to the simulation module need to be implemented (see the flowchart of Figure 11.1). In fact, this corresponds to the inverse procedures mentioned above, which were called $saveXopt()$ and $disaggregateXopt()$. The first procedure is called after each optimisation iteration if the *aggregDecision* switch is set to $false$. The procedure receives the vector of the decision variables from the optimiser ($\mathbf{X}_{\text{opt}}$) and replaces the new values of time rates and speeds in the corresponding positions of the TIME controls (see Table 11.1 for both pumps and valves and of the speed patterns for all variable-speed pumps being optimised. Since the values in the vector of decision variables are normalised, they are converted for the real values.

In case of aggregation of variables, the procedure *disaggregateXopt()* is called instead. This procedure has similarities with the saveXopt() procedure since the objective is also to save the variables that came from the optimiser in the model of the network (replacing the patterns and controls) in order to execute an hydraulic simulation. However, in this case, the variables are aggregated, meaning that the steps need to be restored to their original form. Thus, the value of a variable that corresponds to an aggregated block should be placed in each step associated to the corresponding block. This is performed using the correspondence vector (**correspMatrix**) such as the demonstrated in Algorithm 19.

Thereafter all the procedures implemented, the optimisation module is capable of performing multiple iterations until reach the stopping criteria, which were defined in this work as a maximum number of iterations/generations.

After disaggregation of the blocks, the value of a pump operating time is introduced in the corresponding pattern in a way to start operating in the beginning of each time-step that constitutes the aggregated block. The problem is that, when the value of the operating time is inferior to the size of the time-step, the pump will be forced to shut-off every single step, as in the example of Figure 11.4a. This causes a large number of pump switches which is not recommendable due to the additional maintenance costs usually associated to excessive switches.



(a) (b)

Figure 11.4: Example demonstrating the procedure of moving the disaggregated steps in order to reduce the number of pump switches.

A new procedure was then implemented in the end of the optimisation process with the aim of reducing the number of pump switches. Algorithm 20 provides the implemented steps for such procedure. At a first stage, the optimal operational solution provided by the optimisation module is saved for a posterior use. After that, a procedure to move the disaggregated steps with values inferior

Algorithm 19 Disaggregation of $\mathbf{X}_{\text{opt}}$ to the control variables.

procedure DISAGGREGATEXOPT()
  Initialise k and j;
  for All p pumps to optimise **do**
    Define and initialise variable to memorise previous value: $memPrevValue = 0$;
    Define a temporary vector, **temp**, of size n_{steps};
    for All s steps **do**
      if $correspMatrix_k = 0$ **then** Set $temp_s = memPrevValue$;
      else if $correspMatrix_k > 0$ **then**
        Obtain the pump operating time: $temp_s = x_{\text{opt},j} \times t_{\text{step}}$;
        Memorise operating time: $memPrevValue = temp_s$;
        Go to next position of the decision variables matrix: $j = j+1$ (from the $(p,j)^{\text{th}}$ to the $(p,j+1)^{\text{th}}$ element of the matrix);
      end if
      Go to next position of the corresponding matrix: $k = k+1$ ($(p,k+1)^{\text{th}}$ element of the matrix);
    end for
    Save pump p operating times: $t_{\text{op},p} = temp_s$;
    if Pump p is variable-speed type **then**
      for All s steps **do**
        if $correspMatrix_k = 0$ **then** Set $temp_s = memPrevValue$;
        else if $correspMatrix_k > 0$ **then**
          Obtain the pump speed: $temp_s = x_{\text{opt},j} \times (M_{\max} - M_{\min}) + M_{\min}$;
          $memPrevValue = temp_s$;
          Go to next position of the decision variables matrix: $j = j+1$ ($(p,j+1)^{\text{th}}$ element of the matrix);
        end if
        Go to next position of the corresponding matrix: $k = k+1$ ($(p,k+1)^{\text{th}}$ element of the matrix);
      end for
      Save pump p speed pattern: $M_p = temp_s$;
    end if
  end for
  for All v valves to optimise **do**
    Define and initialise variable to memorise previous value: $memPrevValue = 0$;
    Define a temporary vector, **temp**, of size n_{steps};
    for All s steps **do**
      if $correspMatrix_k = 0$ **then** Set $temp_s = memPrevValue$;
      else if $correspMatrix_k > 0$ **then**
        Obtain the valve opening time: $temp_s = x_{\text{opt},j} \times t_{\text{step}}$;
        Memorise opening time: $memPrevValue = temp_s$;
        Go to next position of the decision variables matrix: $j = j+1$ ($(p,j+1)^{\text{th}}$ element of the matrix);
      end if
      Go to next position of the corresponding matrix: $k = k+1$ ($(p,k+1)^{\text{th}}$ element of the matrix);
    end for
    Save valve v opening time: $t_{\text{op},v} = temp_s$;
  end for
end procedure

to the size of the time-step (*moveAggregSteps*()) is performed in order to group the pump operating steps and hence, reduce the number of pump switches, as demonstrated in Figure 11.4b. A drawback of this procedure is that moving the period of operation of the pumps will also cause influence in the water levels variation in tanks, which, eventually, may lead to the violation of the tanks levels constraints. For this reason, an analysis on the tanks water levels variations with the changes in the pumps operations is performed in the *moveAggregSteps*() procedure. Furthermore, after moving the pumps periods of operation, the new solution is always evaluated by calling the function *updateOF*() that returns the updated value of the cost function (objective function).

Algorithm 20 Reduction of pump switches after the optimisation process.

```
1:  procedure REDUCENPUMPSWITCHES()
2:      if aggregDecision = true then
3:          for All pumps and valves to optimise do
4:              Save the solution (speed/status and time rate) obtained by the optimiser;
5:          end for
6:          Search for other positions for the steps that resulted from the aggregation in order to
            reduce the number of pumps switches: moveAggregSteps();
7:          Compute the cost function with the updated solution (grouped operating times - reduced
            number of switches): updateOF();
8:          if Updated solution maintains or reduces the OF then
9:              Replace the solution by the updated one (grouped operating times);
10:         else
11:             Maintain the initially saved solution;
12:         end if
13:     end if
14: end procedure
```

The *updateOF*() function follows exactly the same steps of Algorithm 6, excluding steps 2 to 6, which correspond to the preparation of variables obtained from the optimiser. In this case, the speed/status variables only need to be updated and evaluated to verify if the new steps positions provide feasible solutions. In case the value of the cost function is maintained (or reduced), this means that the constraints were satisfied and the new solution is accepted. Otherwise, such solution is replaced by the initially saved solution that was obtained from the optimisation module.

The function that performs the changes in the disaggregated steps positions, *moveAggregSteps*(), starts by searching the periods where blocks aggregations were performed ($AggregStep_s > 1$) and obtains the number of steps aggregated (step 11 in Algorithm 21). The values of water levels in each hour for all tanks that were saved during the simulation of the network are then used in this stage. The differences between the levels in consecutive hours are computed (step 15) and then, it is verified if the water levels boundaries are overcome. The solution of grouping blocks in the left is considered feasible for that tank if the tank level constraint is not violated. The process is repeated for all blocks of all pumps and valves.

Algorithm 21 Reduction and movement of aggregated steps.

1: **procedure** MOVEAGGREGSTEPS()
2: **for** All pumps to optimise **do**
3: Obtain $\mathbf{t}_{\text{op}}$ of pump;
4: Define a vector for updated time rates, $\mathbf{t}_{\text{op}}^{\text{new}}$;
5: **for** All s steps of the **AggregStep** matrix **do**
6: Define a temporary variable (temp);
7: **if** $AggregStep_s = 1$ **then**
8: Set $temp = t_{\text{op},s}$;
9: Set $t_{\text{op},s}^{\text{new}} = temp$;
10: **else if** $AggregStep_s > 1$ **then**
11: Obtain the number of aggregated steps: $n_{\text{aggregSteps}} = AggregStep_s$;
12: Initialise a variable to count feasible solutions for each tank: $tankOK = 0$;
13: **for** Each t tank **do**
14: Obtain vector of tank levels, **tankLevels**, of size $n_{\text{steps}} + 1$ (previously saved during the hydraulic simulation for all hours);
15: Compute levels difference between consecutive hours and save to vector **levelsDiff** (of size n_{steps});
16: **if** $L_{\min} < \left[tankLevels_s + (levelsDiff_s \times n_{\text{aggregSteps}})\right] < L_{\max}$ **then**
17: Set $tankOK = tankOK + 1$;
18: **end if**
19: **end for**
20: **if** $tankOK = n_{\text{tanks}}$ **then**
21: Set $temp = t_{\text{op},s} \times n_{\text{aggregSteps}}$;
22: **if** $temp < t_{\text{step}}$ **then**
23: Set $temp = t_{\text{op},s}^{\text{new}}$ and fill the following $n_{\text{aggregSteps}} - 1$ positions of $t_{\text{op}}^{\text{new}}$ with zeros;
24: **else if** $temp > t_{\text{step}}$ **then**
25: **for** All $n_{\text{aggregSteps}}$ **do**
26: **while** $temp > t_{\text{step}}$ **do**
27: Set $t_{\text{op},s}^{\text{new}} = t_{\text{step}}$;
28: Set $temp = temp - t_{\text{step}}$;
29: Go to next $n_{\text{aggregSteps}}$;
30: **end while**
31: **if** $temp > 0$ **then** Set $t_{\text{op},s}^{\text{new}} = temp$;
32: **else** Set $t_{\text{op},s}^{\text{new}} = 0$;
33: **end if**
34: **end for**
35: **end if**
36: **end if**
37: Set $s = s + n_{\text{aggregSteps}} - 1$;
38: **end if**
39: **end for**
40: **end for**
41: **for** All valves to optimise **do**
42: Obtain $\mathbf{t}_{\text{op}}$ of valve;
43: Repeat steps 4 to 39;
44: **end for**
45: **end procedure**

11.3 Energy recovery module

The module devoted to the search for locations in a water network with meaningful potential for energy recovery consists essentially in three main steps:

A. Find locations with significant available hydraulic power;
B. Select/design adequate types of turbines for such locations;
C. Perform a preliminary feasibility analysis.

The implemented procedures to perform step A are described in Algorithms 22 and 23. In the procedure listed in the algorithm, an hydraulic simulation of the network under analysis is performed. The obtained values of flow rate and head drop in each link for all time periods of the simulation are used to compute the variation and total hydraulic power available in each potential site.

Algorithm 22 Search for potential locations for energy recovery in a network.

```
1:  procedure SITESLOCATION1()
2:      Open toolkit: ENopen(<network>.inp);
3:      Open hydraulic solver: ENopenH();
4:      Initialise hydraulic analysis: ENinitH(00);
5:      while time period > 0 do
6:          Run hydraulic analysis: ENrunH(clockTime);
7:          for each clockTime do
8:              for All k links do
9:                  Get the value of flow rate, Q, obtained: ENgetlinkvalue();
10:                 if Link k is a pipe then
11:                     Identify the input and output nodes of the pipe, nodeIN and nodeOUT:
                        ENgetlinknodes();
12:                     Compute the site head drop, H = H_nodeIN − H_nodeOUT;
13:                 else
14:                     Get the value of valve headloss, which corresponds to the link head drop, H,
                        through the link: ENgetlinkvalue();
15:                 end if
16:                 Compute the available hydraulic power P = γQH;
17:             end for
18:         end for
19:         Calculate the length of the next timePeriod: ENnextH(timePeriod);
20:     end while
21:     Close hydraulic solver: ENcloseH();
22:     Close toolkit: ENclose();
23:     for All k links do
24:         Compute the minimum and the mean values of flow, Q_min and Q_mean, and head, H_min and
            H_mean;
25:         Sum the values of hydraulic power obtained in each time period;
26:     end for
27:     Save all information of flow, head and power to a file sorting from the highest to the lowest
        power.
28: end procedure
```

Algorithm 23 Implementation of virtual pressure reducing valves (PRV) and calculation of the potentially recoverable energy.

1: **procedure** SITESLOCATION2()
2: Define and initialise the number of scenarios: $n_{\text{scen}} = 1$;
3: **for** All p pipes **do**
4: Open <network>.inp file;
5: Add a pressure reducing valve (PRV) in the output node of pipe p;
6: Set $K_{\text{m}} = 0$;
7: Save the file with the name: <network>Sn_{scen}.inp;
8: Find the coefficient, K_{m}, that maximises the energy production: $optimalValveCoeff()$;
9: Save the value of potentially recoverable energy, E_{recov}, of the current scenario to a vector;
10: Go to next scenario: $n_{\text{scen}} = n_{\text{scen}} + 1$;
11: **end for**
12: Save the sorted values of recoverable energy (from the highest to the lowest), as well as the corresponding values of flow and head (including the minimum and mean) for all scenarios to file;
13: **end procedure**

The procedure listed in Algorithm 23 consists in the analysis of each potential site by installing a virtual pressure reducing valve (PRV) in the end of each pipe. The minor headloss coefficient initially attributed to the PRV is $K_{\text{m}} = 0$. For each scenario (1 scenario per site/pipe), a search for the optimal value of K_{m} is performed by calling the $optimalValveCoeff()$ function. This function, described in Algorithm 24, (i) replaces, in each iteration, the loss coefficient K_{m} by a new value, (ii) performs an hydraulic simulation of the network to compute the energy dissipated by the valve and (iii) obtain the new values of pressure at the demand nodes. If the nodal pressures do not violate the imposed constraint (see the optimisation problem formulation in Equation 10.8 and the computed value of dissipated energy is maintained or increased, then the changes performed in the network model are saved and a new value of K_{m} is tested. Otherwise, the previous value of K_{m} is maintained, returning the corresponding dissipated energy (the potentially recovered energy).

The second main step executed in the energy recovery module, described in Algorithm 25, consists in the selection of the types of turbines that are adequate to each different scenario. The procedure selects the turbines according to the head ranges specified by the European Small Hydropower Association (2004), as presented in Table 10.1. A vector containing the information concerning the selected turbines is saved for each scenario/site.

The last main step of the energy recovery module consists in two procedures: (i) assess the technical feasibility and (ii) evaluate the financial feasibility of the selected turbines, whose pseudo-code description is presented in Algorithms 26 and 27, respectively. The first procedure is essentially devoted to the design of each type of turbine selected for each different scenario. The turbines' diameter is computed, as well as the specific speed for reaction turbines. Afterwards, the efficiency of each turbine is computed according to the defined efficiency equations for each type of turbine (Equations 10.10 to 10.22), as well as the resulting net recovered power, P_{net}.

The procedure of financial feasibility provides, for each scenario, (i) the annual revenue predicted using each different type of turbine associated to each scenario, (ii) the total investment, including the cost of the turbine and its installation and the cost of the civil works, and, finally, (iii) the payback

Algorithm 24 Search for the minor loss coefficient of a pressure reducing valve (PRV) that maximises the dissipated energy.

function OPTIMALVALVECOEFF()
  Define $E_{\text{recov}}^{\text{initial}} = 0$;
  Open toolkit: $ENopen$(<network>Sn_{scen}.inp);
  Open hydraulic solver: $ENopenH()$;
  while time period > 0 **do**
    Run hydraulic analysis: $ENrunH(clockTime)$;
    for each $clockTime$ **do**
      for the link corresponding to the added PRV **do**
        Save the value of flow rate, Q, and headloss, h_{m}, obtained: $ENgetlinkvalue()$;
        Compute the dissipated energy that can be recovered, E_{recov};
        Set variable $E_{\text{recov}}^{\text{initial}} = E_{\text{recov}}^{\text{initial}} + E_{\text{recov}}$;
      end for
    end for
    Calculate the length of the next time period: $ENnextH(timePeriod)$;
  end while
  Close hydraulic solver: $ENcloseH()$;
  Close toolkit: $ENclose()$;
  Define variables $\phi d = 500$ and $E_{\text{recov}}^{\text{new}} = 0$;
  while $K_{\text{m}} \leq 2 \times 10^5$ **do**
    Set $K_{\text{m}} = K_{\text{m}} + \phi d$;
    Open toolkit: $ENopen$(<network>Sn_{scen}.inp);
    Open hydraulic solver: $ENopenH()$;
    Set the new value of loss coefficient K_{m} in the PRV: $ENsetlinkvalue()$;
    while $timePeriod > 0$ **do**
      Run hydraulic analysis: $ENrunH(clockTime)$;
      for each $clockTime$ **do**
        for the link corresponding to the added PRV **do**
          Get the value of flow rate, Q, and headloss, h_{m}, obtained: $ENgetlinkvalue()$;
          Compute the dissipated energy that can be recovered, E_{recov};
          Set variable $E_{\text{recov}}^{\text{new}} = E_{\text{recov}}^{\text{new}} + E_{\text{recov}}$;
        end for
        for Each demand node **do**
          Get the value of pressure: $ENgetnodevalue()$;
        end for
      end for
      Calculate the length of the next time period: $ENnextH(timePeriod)$;
    end while
    if Pressure at demand nodes ≥ 25 m and $E_{\text{recov}}^{\text{new}} \geq E_{\text{recov}}^{\text{initial}}$ **then**
      Set $E_{\text{recov}}^{\text{initial}} = E_{\text{recov}}^{\text{new}}$;
      Save updated model of the network: $ENsaveinpfile$(<network>Sn_{scen}.inp);
      Close hydraulic solver: $ENcloseH()$;
      Close toolkit: $ENclose()$;
    else
      Close hydraulic solver: $ENcloseH()$;
      Close toolkit: $ENclose()$;
      Break the while-cycle and go to next step;
    end if
  end while
  return $E_{\text{recov}} = E_{\text{recov}}^{\text{initial}}$.
end function

Algorithm 25 Selection of the adequate turbine type for each site.

```
1: procedure TURBINESELECTION()
2:     Open file with site information;
3:     for each scenario n_scen do
4:         Read the values of H_min and H_mean;
5:         Define a vector to save the selected types of turbine: selectedTurb;
6:         Go to the end of the file;
7:         if H_mean > 50 then
8:             Write in file "Scenario Sn_scen: Pelton, Turgo, Francis, cross-flow or PAT";
9:             Save to a vector the selected turbines: selectedTurb=[1 2 3 4] (where 1=Pelton,
               2=Turgo, 3=Francis, 4=cross-flow, 5=Kaplan and 6=propeller);
10:            Go to next scenario;
11:        else if H_min > 10 and H_mean > 40 then
12:            Write in file "Scenario Sn_scen: Francis, cross-flow or PAT";
13:            Save to a vector the selected turbines: selectedTurb=[3 4];
14:            Go to next scenario;
15:        else if H_min > 10 then
16:            Write in file "Scenario Sn_scen: Francis, cross-flow, Kaplan/propeller or PAT";
17:            Save to a vector the selected turbines: selectedTurb=[3 4 5 6];
18:            Go to next scenario;
19:        else if H_min > 3 then
20:            Write in file "Scenario Sn_scen: cross-flow, Kaplan/propeller or PAT";
21:            Save to a vector the selected turbines: selectedTurb=[4 5 6];
22:            Go to next scenario;
23:        else if H_min > 2 and H_mean < 40 then
24:            Write in file "Scenario Sn_scen: Kaplan/propeller or PAT";
25:            Save to a vector the selected turbines: selectedTurb=[5 6];
26:            Go to next scenario;
27:        else
28:            Write in file "Scenario Sn_scen: site not adequate for energy recovery";
29:            Save the value 0 to the vector of selected types of turbine: selectedTurb=[0];
30:            Go to next scenario;
31:        end if
32:    end for
33:    Close file.
34: end procedure
```

Algorithm 26 Assess the technical feasibility of the selected turbines.

```
1: procedure TECHNICALFEASIBILITY()
2:     for each scenario n_scen do
3:         Compute the gross power, P_gross, according to Equation 10.5;
4:         for all i positions of the vector selectedTurb do
5:             if selectedTurb_i = 0 then
6:                 Break the cycle and go to next scenario;
7:             else if selectedTurb_i =3 or 5 (reaction turbines) then
8:                 Compute the runner diameter, D_t, according to Equation 10.10, where Q_d is the
                   Q_mean;
9:                 if selectedTurb_i =3 (Francis) then
10:                    Compute the specific speed, N_Q, and the turbine efficiency according to Equa-
                       tions 10.11 and 10.12;
11:                else if selectedTurb_i =5 (Kaplan) then
12:                    Compute the specific speed, N_Q, and the turbine efficiency according to Equa-
                       tions 10.11 and 10.15;
13:                else if selectedTurb_i =6 (propeller) then
14:                    Compute the specific speed, N_Q, and the turbine efficiency according to Equa-
                       tions 10.11 and 10.17;
15:                end if
16:            else if selectedTurb_i = 1 (Pelton) then
17:                Compute the turbine efficiency according to Equation 10.18;
18:            else if selectedTurb_i = 2 (Turgo) then
19:                Compute the turbine efficiency according to Equation 10.21;
20:            else if selectedTurb_i = 4 (cross-flow) then
21:                Compute the turbine efficiency according to Equation 10.22;
22:            end if
23:            Compute the net power, P_net, according to Equation 10.6;
24:        end for
25:    end for
26: end procedure
```

time for each type of turbine.

Algorithm 27 Assess the financial feasibility of the selected turbines.

procedure FINANCIALFEASIBILITY()
  for each scenario n_{scen} **do**
    Compute the annual recovered energy: $E_{annual} = E_{net} \times 365$;
    Compute the annual revenue: $revenue = E_{annual} \times energySellPrice$;
    for all i positions of the vector **selectedTurb do**
      if $selectedTurb_i = 1$ (Pelton) or 2 (Turgo) **then**
        Compute the turbine cost according to Equation 10.23d;
      else if selectedTurbVect(i)=3 (Francis) **then**
        Compute the turbine cost according to Equation 10.23a;
      else if selectedTurbVect(i)=4 (cross-flow) **then**
        Compute the turbine cost according to Equation 10.23e;
      else if selectedTurbVect(i)=5 (Kaplan) **then**
        Compute the turbine cost according to Equation 10.23b;
      else if selectedTurbVect(i)=6 (propeller) **then**
        Compute the turbine cost according to Equation 10.23c;
      end if
      Compute the turbine installation cost and the cost of civil works according to Equations 10.24 and 10.25, respectively;
      Define a variable $year = 0$;
      Set $cashFlow = C_t + C_{instal} + C_{civil}$;
      while $cashFlow \leq 0$ **do**
        $cashFlow = cashFlow - revenue$;
        $year = year + 1$;
      end while
      $PaybackTime = year$;
    end for
  end for
end procedure

References

EPA U.S. (2015). *Water research: EPANET.* Retrieved from http://www2.epa.gov/water-research/epanet

European Small Hydropower Association. (2004). Guide on How to Develop a Small Hydropower Plant (Part 2).. Retrieved 2015, from https://energypedia.info/images/4/4a/Part_2_guide_on_how_to_develop_a_small_hydropower_plant-_final-21.pdf

Hutt, M. F. (2007). *Michael Hutt's Home Page - Nelder-Mead Simplex Method.* Retrieved 2015, from http://www.mikehutt.com/crosen.c

Ingber, L. (2015). *Lester Ingber's Archive - ASA.* Retrieved 2015, from http://www.ingber.com/#ASA

López-Ibáñez, M. (2015). *EPANET Programmer's Toolkit (help manual).* Retrieved from http://lopez-ibanez.eu/doc/toolkit_help.pdf

Microsoft: developer network. (2015). *Path.gettempfilename method ().* Retrieved from https://msdn.microsoft.com/en-au/library/system.io.path.gettempfilename.aspx

Rossman, L. A. (2000). *Epanet 2: users manual.* US Environmental Protection Agency. Office of Research and Development. National Risk Management Research Laboratory.

Salomons, E. (2015a). *Water simulation: DXF2EPA.* Retrieved from http://www.water-simulation.com/wsp/2005/06/03/dxf2epa-autocad-dxf-file-conversion-utility-for-epanet/

Salomons, E. (2015b). *Water simulation: EPACAD.* Retrieved from http://www.water-simulation.com/wsp/2010/08/25/epacad/

Takahama, T. (2005a). *Sample program of Differential Evolution.* Retrieved 2015, from http://www.ints.info.hiroshima-cu.ac.jp/~takahama/download/DE.html

Takahama, T. (2005b). *Sample program of Particle Swarm Optimisation.* Retrieved 2015, from http://www.ints.info.hiroshima-cu.ac.jp/~takahama/download/PSO.html

Takahama, T. (2005c). *Sample program of Real Coded Genetic Algorithm.* Retrieved 2015, from http://www.ints.info.hiroshima-cu.ac.jp/~takahama/download/GA.html

Zonum Solutions. (2009). *Zonum Solutions: epa2GIS.* Retrieved from http://www.zonums.com/epa2gis.html

Part V

Validation, results and discussion

12. Modelling networks under distinct operational and design conditions

The use of variable-speed pumps in water supply systems is analysed. Modelling results of a single-pump network considering distinct pump efficiencies, distinct pump speeds and operating times are presented. The same network is also changed in terms of geometric head and pipes roughness and the modelling results are compared. The proposed formula for computing the efficiency of variable-speed pumps is also tested.

There is a lack of studies quantifying the effects of using variable frequency drives (VFD) in the efficiency improvement of water pumps operation. At the same time, several users of simulation computer programmes may not have information on the approximations considered in each distinct programme concerning the pumps efficiency curves modification for distinct speeds of operation. This means that, when performing, for instance, a cost analysis for the installation of variable frequency drives (VFD), the results will also be affected by those approximations.

It is also known that the convenience of using variable-speed pumps can be different depending on the network dimensions and configurations. Therefore, it is important to quantify the implied differences. In this chapter, systems presenting distinct characteristic head loss curves, and hence distinct behaviours, are tested with a variable-speed pump. The economic and energetic benefits of a variable-speed pump when compared with the use of a fixed-speed pump are quantified.

Several water utilities* have already decided to install variable-speed drives, trying to improve the efficiency of their systems. However, the variable-speed pumps are always set to operate at the best efficiency point (BEP). No advantages are taken from the possibility of operating at distinct speeds. This is mostly happening due to the so difficult task of selecting the adequate speeds of operation capable of improving the network efficiency and, at the same time, maintaining the minimum requirements for the consumers. This chapter also intends to demonstrate the potential of varying the speed of pumps in certain periods of operation instead of considering a fixed speed value.

A list of the modelling conditions presented in this chapter with the aim of analysing the energetic and economic savings from different uses of variable-speed pumps is provided:

- Pumps efficiency - pump with constant efficiency or with an associated efficiency curve are analysed and compared;

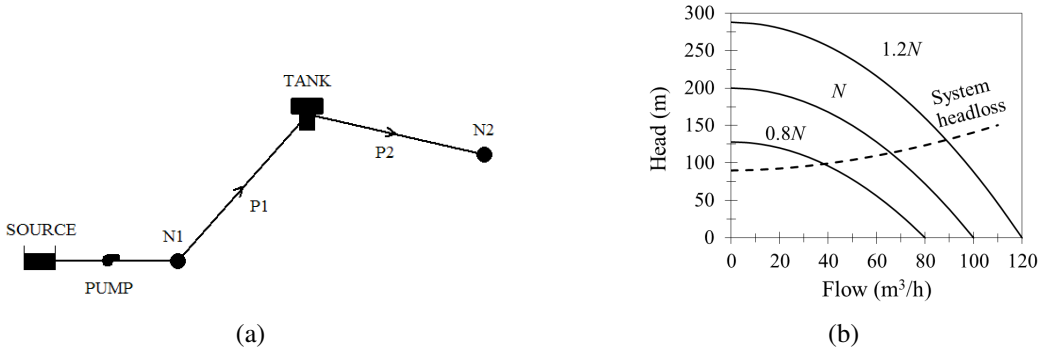
*This statement is based on contacts made directly with professionals in the field, *i.e.* working for or in water utilities.

- Pumps speed - the advantages of using a pump with variable speed instead of constant speed are analysed;
- Pumps operating times - distinct operating times for the pump operation instead of the common fixed 1-hour steps are used when modelling the network operation, both at variable- and fixed-speed;
- Multiple system head loss curves - testing the effect of using variable-speed pumps in systems with distinct head loss curves;
- Pumps efficiency computation - distinct formulae for the pump efficiency prediction are analysed.

To perform all the analyses, the single-pump network was modelled and simulated using Microsoft Excel. The developed Excel-based tool, available in the GRIDS research group website (*GRIDS: Water GRIDS*, 2015), follows the same methodology of EPANET 2.0. Such tool allowed to easier perform changes in the network operational and design parameters and to automatically obtain the results in the required graphical format. At the same time, it allowed a detailed analysis on the EPANET methodologies and a comparison of results (Coelho & Andrade-Campos, 2016b).

12.1 Description of the single-pump network

The simple network analysed in this chapter (see figure 12.1a) is composed of a water source, a storage reservoir (or tank) that supplies the point of consumption represented by node N2 (with an associated consumption pattern) and one pump responsible for pumping the water from the source to the storage tank.



(a) (b)

Figure 12.1: (a) EPANET representation of the single-pump network used in this work and corresponding (b) system head loss curve and pump characteristic curves (at nominal speed, N, and at lower and higher relative speeds, $0.8N$ and $1.2N$, respectively).

The elevations of the node N1, the tank and node N2 are, respectively, 10, 100 and 90 m and the total head of the water source is 10 m. The tank has a diameter of 5 m and their minimum and maximum levels of operation are 2 and 20 m, respectively. The pump characteristic head curve as well as the corresponding curves for the pump at a lower and a higher relative speeds (80 % and 120 % of the nominal speed) and the curve of the network pipes head loss are provided in figure 12.1b. The base demand of node N2 is 10 m^3/h and the associated demand pattern is provided in figure 12.2 which also includes the pattern for the energy price variation during a representative day.

The pipe that links node N1 to the tank (pipe P1) is characterised by a length of 2000 m, a

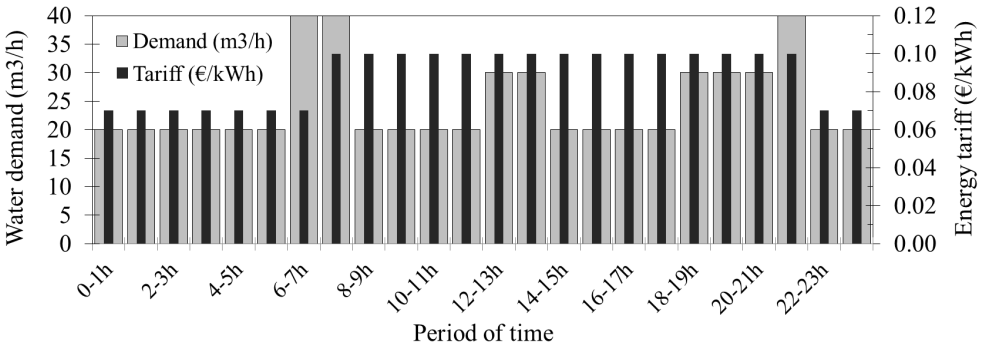


Figure 12.2: Water demand pattern associated to node N2 of the case-study network and the pattern of energy price (tariff) considered for the operational energy costs computation.

diameter of 200 mm and a roughness coefficient of 50. The pipe that links the tank to node N2 (pipe P2) presents exactly the same dimensions and also a roughness coefficient of 50 is considered for the Hazen-Williams head loss calculation. In order to ignore minor head losses in both pipes, the minor head loss coefficient was set to zero.

12.2 Initial modelling conditions

The previously described single-pump network is modelled for a simulation period of 1 day divided into time periods of 1 hour (hourly time-steps).

Initially, for the first hydraulic simulation of the network operation, the time periods are considered fixed, meaning that the pump is only allowed to operate during the entire time-step (*i.e.* never less than 1 hour). At the same time, only the nominal speed of the pump is considered. Figure 12.3 shows the pattern of the pump operation considered in the initial model of the network, as well as the results of the water level in tank during all period of operation.

As observed in figure 12.3, the variation of the energy price during the day is taken into account and the pump is mainly operating during the lower electricity cost periods. Such operational conditions represents already an intuitive attempt to model the most efficient operation of the case-study network (a *trial and error* optimisation procedure was used) considering a pump with no variable speed.

Although results of figure 12.3 correspond to the ones obtained using the Excel-based tool, the same values for the tank water level variation were obtained using EPANET 2.0, as well as flow and pressure results at each step, validating the developed Excel-based tool.

In section 12.3, the previously described pump operational conditions are changed in order to verify the influence of particular changes in the final operating costs when compared with this initial model. Both changes are also tested considering a constant value for the pump efficiency and considering the use of an efficiency curve. Figure 12.4 provides the representation of the efficiency curve considered for the pump and also the pump operating points for each time period of the simulation considering the controls represented by the pump pattern of figure 12.3.

Despite the pump is always operating at the same speed, it is possible to observe that the results for the operating points (also similar to the obtained with EPANET) are slightly deviated. This is caused by the variations of the water level in tank. When the water level is higher, the pump will need

more energy to overcome the elevations difference and then, the corresponding operating point will be deviated to superior values of pumping head.

It should be noted that the tank water level in the end of the simulation period (at 24 hours) is never allowed to be inferior to the initial level, *i.e.* is never lower than the water level in the beginning of the simulation (at 0 hours, in this case).

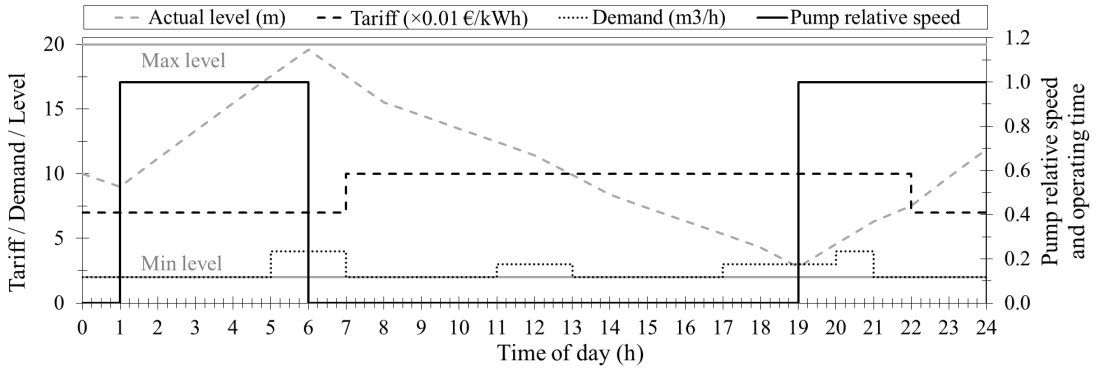


Figure 12.3: Initial pattern considered for the operation of the pump of the case-study network and evolution of the water level in tank during the simulation period considering such pump controls. Comparison with patterns of both water demand and energy price (tariff).

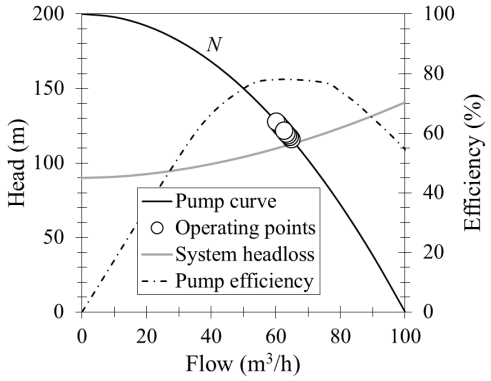


Figure 12.4: Representation of (i) the efficiency curve and (ii) the pump operating points correspondent to the controls considered in the initial model of the case-study network.

12.2.1 Model considering constant efficiency

The first model for the simulation of the case-study network operation considers a pump with a constant efficiency of 75 %.

The daily cost computed by the Excel-based tool has the value of 21.79 €. EPANET presented a total cost of 21.78 €. These first results demonstrate a satisfactory approximation of the developed tool with EPANET, presenting a relative error of 0.063 %.

12.2.2 Model considering an efficiency curve

The second model for the simulation of the case-study network initial operation considers a variation of the pump efficiency with the discharge (pump flow) instead of a constant efficiency. The consid-

ered pump efficiency curve is represented in figure 12.4. In EPANET, the same efficiency curve is associated to the pump.

Considering the same initial conditions for the network operation during the simulation period of one day, the average efficiency of the pump is 77.8 %. The daily cost associated to the pump operation is 21.01 €, computed with the Excel-based tool and 20.99 € computed with EPANET. In this case, the discrepancy between the results obtained with the developed tool and with EPANET resulted in a relative error of 0.088 %.

As expected, since the average pump efficiency is superior to the constant efficiency considered in the previous model, the computed daily cost is consequently inferior.

Besides the difference between the computed costs when compared with the model using constant efficiency, the behaviour of the network, respecting the tank water levels and the pump operating points, follows the same pattern since the controls used for the pump (speed and operating time) are exactly the same.

12.3 Assessing distinct operational conditions

12.3.1 Changing the pump speed

At this stage, the influence of using, for instance, a variable frequency drive (VFD) for changing the pump speed in order to reduce the operational costs is tested. Therefore, a different pattern for the pump operation, considering distinct speeds, is considered. The pump speed pattern considered for the model of this section is represented in figure 12.5. The pump relative speed was reduced to 0.9 from 5 a.m. to 7 a.m. and from 9 p.m. to 10 p.m.. From 7 p.m. to 9 p.m., the speed was reduced to 80 % of the nominal (0.8).

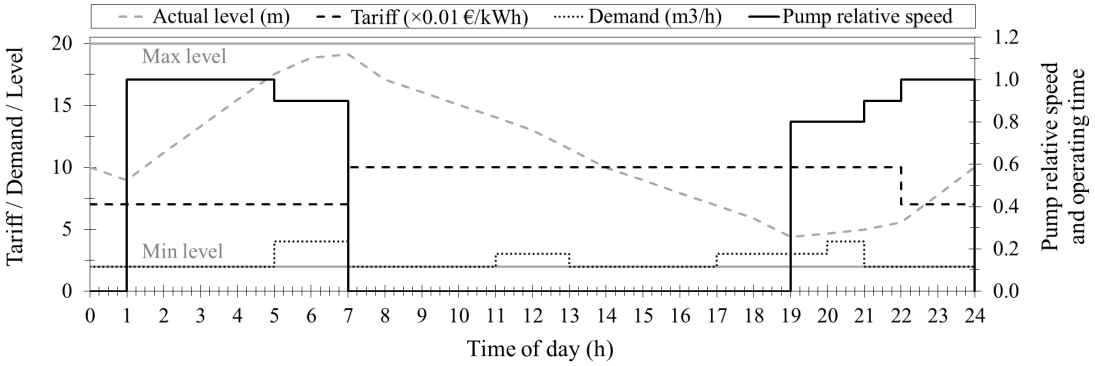


Figure 12.5: Main characteristics of the case-study network model, during the 24 hours simulation period, considering distinct speeds for pump.

Although figure 12.5 shows a similar pattern of the tank water level evolution when compared with the initial model, improvements in the operational costs were obtained by using distinct pump speeds.

Considering the described conditions and a constant pump efficiency of 75 %, the daily operational cost of the case-study network obtained with the Excel-based tool is 19.12 €. For the same model, using EPANET, the obtained cost is 19.10 €, meaning that the relative error of the developed tool is 0.085 %.

When considering the pump efficiency curve, the operational cost of the case-study network has the value of 19.47 € in the developed tool and of 19.45 € in EPANET (a relative error of 0.089 %). The average pump efficiency, in this case, is 73.2 %.

Figure 12.6 shows the pump operating points at each distinct speed considered. Despite the pump is operating at certain time periods in efficiency conditions superior to 75 %, in other periods, the pump operates at lower-efficiency points, resulting in an average efficiency inferior to 75 %. It is also observed that the pump is not operating at the most efficient point (the point where the pump head curve intersects the system characteristic head loss curve) at 90 % of the nominal speed. On the other hand, the use of the pump at lower speeds enable the pump to adapt to the demand variability and operate at smaller flow and head values which, in turn, results in a reduction of the power consumption.

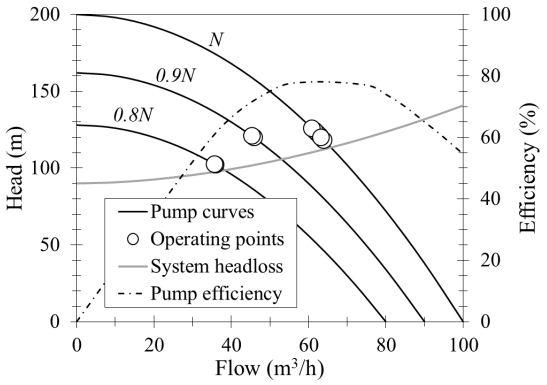


Figure 12.6: Results of the pump operating points for the model considering distinct pump speeds.

The results obtained in this modelling case demonstrate that even without reducing the time of pump operation (in fact, the time of operation has been increased in one hour - from 6 to 7 a.m. - in comparison with the previous case) and without moving the pump operation to cheaper periods of the day, reductions in the operational costs can be obtained by adapting the pump operation to the demand flow variation by changing the rotational speed.

12.3.2 Changing the pump operating times

In this section, distinct operating times are considered for the pump operation instead of considering the fixed 1-hour operating times. The main idea is to reduce the case-study network operational cost by reducing the time of operation required by the pump considered, in this case, of fixed-speed.

The pump pattern, including the distinct operating times used in this case-study, are presented in Figure 12.7. The pump is turned-off at 4 hours, turned-on between 5 and 5.9 hours (54 minutes of operation), it is also turned-on between 6 and 6.7 hours (42 minutes of operation) and only turned-on 52 minutes at 20 hours in order to avoid the tank to reach the minimum level. During the other periods of the pump operation, the operating time was maintained in 1 hour.

Figure 12.7 also provides the results for the tank water levels evolution considering the pump operation using the controls considered in this model. The figure shows a slightly different pattern in the variation of the tank water level caused by the distinct pattern of the pump operation. However, in this case, reductions were also obtained in the operational costs when comparing to the initial model,

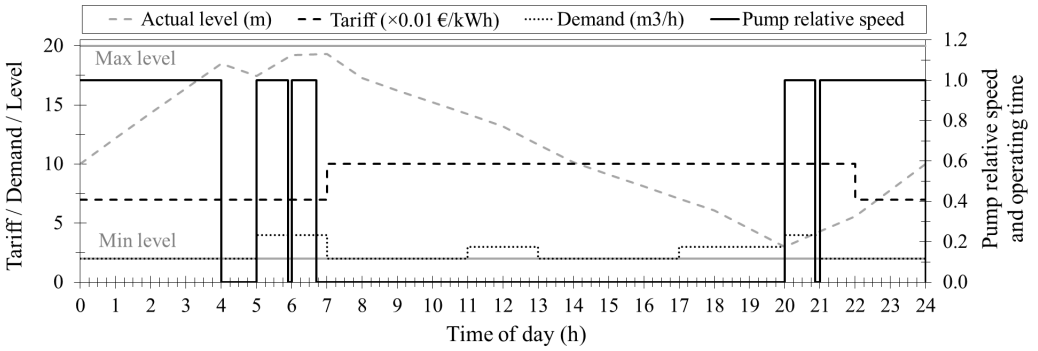


Figure 12.7: Main characteristics of the case-study network model during the 24 hours simulation period considering distinct operating times instead of the fixed 1-hour.

due to minor reduction in the pump operating time.

Considering the pump constant efficiency of 75 %, the values obtained for the network operational cost were 19.82 € in EPANET and 19.83 € in the Excel-based tool (a relative error of 0.067 %).

When considering the pump efficiency curve, with a verified average efficiency of 77.8 %, the values of the operational cost for the case-study network were 19.10 € in EPANET and 19.11 € in the developed tool (a relative error of 0.076 %).

In figure 12.8a is possible to verify the pump operating points resulted from this model considering always the pump nominal speed and distinct operating times, which shows the pump operating at higher efficiency points.

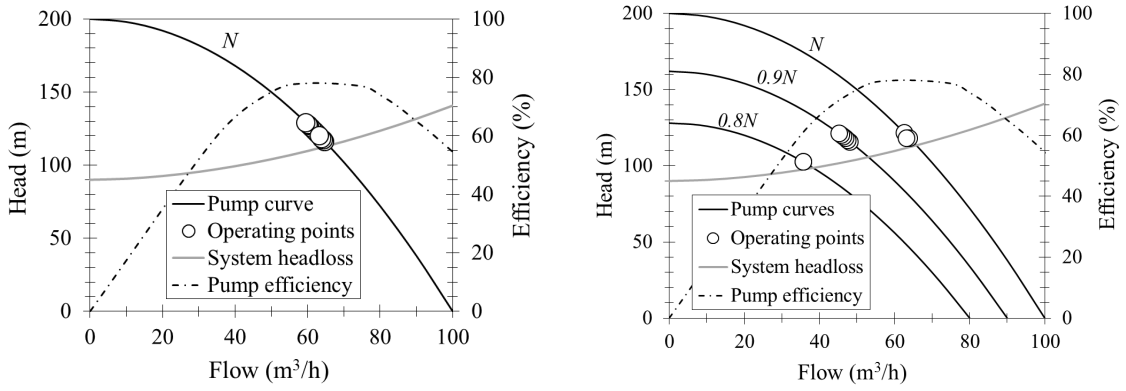


Figure 12.8: Results of pump operating points for models under distinct operating conditions: (a) considering distinct operating times instead of the fixed and (b) considering a variable-speed pump with distinct operating times instead of the fixed 1-hour steps.

Since the initial solution considered for the periods of pump operation already take advantage of the variation of the energy price, it is not possible to quantify the savings obtained by just moving the pump operation to lower cost periods of the day. However, although not evidenced in the presented results, analysing each time-step individually, the savings in operational costs demonstrated, as expected, to be superior when the reduction of the pump operating time is made for periods of higher energy cost.

12.3.3 Changing simultaneously the pump speed and operating times

A model considering simultaneously non-constants pump relative speeds and different operating times is tested in this section. Figure 12.9 shows a possible set of controls for the pump of the case-study network considering distinct pump relative speeds and distinct operating times. In this case, the values for the relative speed varies from 0.8 to 1 while the operating time is only reduced in the $7-8$ p.m. time interval, otherwise the limits of the tank levels would not be respected.

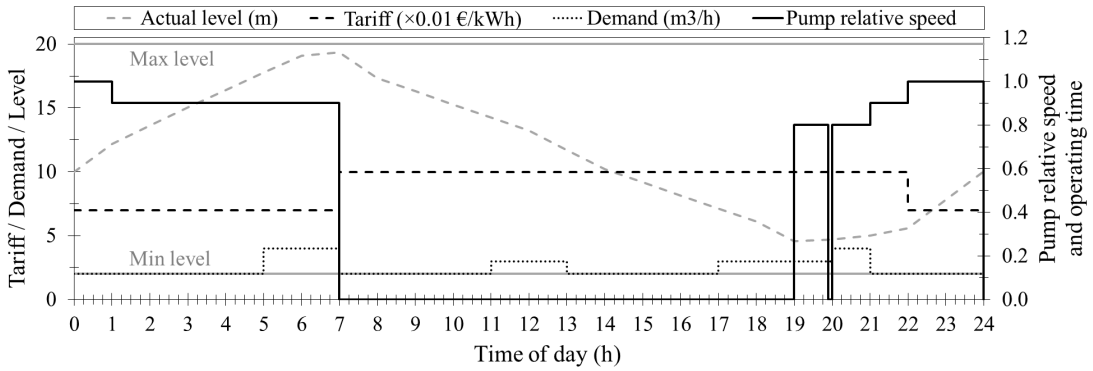


Figure 12.9: Main characteristics of the case-study network model during the 24 hours simulation period considering both distinct pump operating times instead of the fixed 1-hour steps and also distinct pump speeds.

Modelling the pump operation with a constant efficiency of 75 %, the results for the daily operational cost were 18.78€ with EPANET and 18.79 € with the Excel-based tool (a relative error of 0.076 %). On the other side, associating an efficiency curve to the pump, the results demonstrated an average efficiency of 71.8 %, meaning that the power consumption is superior in this case, which increases the costs. The final daily cost obtained is 19.52 € with EPANET and 19.54 € with the developed tool, which presents a relative error of 0.101 %.

Observing also the pump operating points with the system and pump characteristic curves provided in figure 12.8b, it is possible to verify that the pump is operating very near the best efficiency points when operating at full speed, N, or at 80 % of full speed ($0.8N$). However, when the relative speed is 0.9, the pump is operating at slightly lower efficiency points, which decreases the average efficiency.

12.3.4 Discussing the distinct operational conditions

In a simple network such as the one presented in this case-study, any experienced operator trying to reduce the operational costs will intuitively opt by a kind of network control following the patterns that were already observed in figures 12.3, 12.5, 12.7 and 12.9. In other words, a pump will be operating essentially during the lower energy cost periods, meaning that the tank will be always emptying during the higher cost period until reach the minimum level.

Despite such kind of intuitive controls seems to provide the most efficient operation of the network, this is not necessarily always true. The tests provided in the previous sections demonstrate that, even a network already optimised by a *trial and error* methodology, with some simple changes in the operational conditions at certain time periods can improve its economical and energetic efficiency.

Tables 12.1 and 12.2 present a resume of the main economical and energetic results obtained with the case-study network considering distinct models based on particular changes in the pump operational conditions and using, respectively, the pump constant efficiency and the efficiency curve.

Table 12.1: Results obtained by changing the operational conditions of the case-study network, considering a constant efficiency pump.

Model	Initial	Speed changes	Time changes	Speed and time changes
Avg efficiency (%)			75.00	
Max Power (kW)	27.89	27.84	27.92	27.65
Daily cost (€)	21.79	19.12	19.83	18.79
Daily energy (kWh)	276.13	252.92	261.54	248.90
Avg energy (kWh/m^3)	0.44	0.42	0.45	0.42
Pumped water (m^3)	627.7	590.0	590.5	590.5
% Cost reduction	-	12.29	9.00	13.76
% Energy reduction	-	8.41	5.29	9.86
% Avg energy reduction	-	3.64	-1.34	4.96
% Water reduction	-	6.00	5.93	5.93

Table 12.2: Results obtained by changing the operational conditions of the case-study network, considering a pump efficiency curve.

Model	Initial	Speed changes	Time changes	Speed and time changes
Avg efficiency (%)	77.82	73.25	77.83	71.78
Max Power (kW)	26.82	26.79	26.88	26.65
Daily cost (€)	21.01	19.47	19.11	19.54
Daily energy (kWh)	266.14	255.34	252.02	257.07
Avg energy (kWh/m^3)	0.42	0.44	0.43	0.44
Pumped water (m^3)	627.7	590.0	590.5	590.5
% Cost reduction	-	7.34	9.02	6.99
% Energy reduction	-	4.06	5.30	3.41
% Avg energy reduction	-	-2.66	-0.83	-3.30
% Water reduction	-	6.00	5.93	5.93

The first main conclusion that can be taken from the results of both tables is that all the operational changes tested in the previously presented models deserve attention since, in all cases, significant economical and energetic improvements can be obtained. In other words, reductions between 6 and 13 % for the daily costs and between 3 and 9 % for the daily energy consumption can be achieved. Such values, considering an entire year operating a network, can represent substantial savings.

Another conclusion that can be taken is related to the pump efficiency considerations when modelling the operation of a pipe network. Considering a pump efficiency curve, which is a better approximation of a real pump, the percentage of cost and energy reduction is inferior when comparing with the same models using a constant efficiency.

Observing the models considering changes in the pump relative speed, when the efficiency curve in used (table 12.2), the values of the average efficiency are lower in these cases, implying more power consumption and, consequently, more costs. The lower average efficiencies when reducing the pump relative speeds are also the reason for the inferior percentage of reduction in energy consumption

and costs. At the same time, in the tests when only the pump operating times are changed, the same percentage of costs and energy reduction are obtained when comparing the cases with constant efficiency and with the efficiency curve. This observation is due to the non-existence of changes in the pump efficiency between the initial case and the case with time changes.

Still discussing the models where small reductions in the time of pump operation resulted in a significant decrease of both energy consumption and costs, it is important to mention here the potential role of using automatic control systems. Since a measly amount of time - let's say, for instance, 15 minutes - of pump operation can result in significant additional energy costs and since this kind of situation can easily occur in systems manually controlled by an operator due to, for example, a larger response time of the operator or even a delay in the control decisions receiving, then, in this situations, a type of automatic control (receiving the control decisions and immediately processing such decisions) can avoid unnecessary operational costs and even possibly save an amount of energy.

The simultaneous application of pump relative speed and operating times changes does not necessarily provides the lowest value of operational cost. Although considering a constant pump efficiency, the minimum cost were obtained for the model with speed and time changes (table 12.1), the same is not observed when considering an efficiency curve for the same pump operational controls since, in this case, the average efficiency presented the lowest value (table 12.2). However, it should be noticed that the controls options being analysed and assessed may not be the most adequate - examples of intuitive solutions were chosen, which cannot represent optimal solutions since no optimisation techniques were applied.

Other important observations are related to the volume of water pumped. Both table 12.1 and table 12.2 show that the changes in the initial operation of the network induced a reduction of approximately 6 % of the total water pumped. It should be noted that a decrease in the operating costs can only be obtained by reducing the volume of water to be pumped or/and by moving the pump operation to more favourable periods of the day according to the energy tariff. On the other hand, an higher volume of water pumped does not necessarily means more energy costs. In table 12.2, comparing the model with speed changes with the model with time changes, it is observed that in the model with time changes, a larger volume of water is pumped (590.5 m^3), however, since the average efficiency is higher, the associated energy consumption and cost is lower. If, on the other side, the efficiency is kept the same (table 12.1), an increase in the volume of water pumped will always imply larger operational costs.

As can be seen from table 12.2, which provides negative values for the percentage of average energy reduction per cubic meter (meaning an increase), the reduction of the daily energy consumption may not imply lower energy consumption per cubic meter of water pumped. This only occurs in the two cases of table 12.1 when the pump speed is reduced, since there is no influence in the pump efficiency.

Both analysed results demonstrate that the approximation of the pump efficiency to a constant efficiency can provide a false perception of the real savings from the use of a variable-speed pump instead of a fixed-speed type. When contrasting with the results obtained by considering a pump efficiency variable with the pumped flow, it is observed that savings in energy and operational costs are smaller but still meaningful.

12.4 Assessing the effect of distinct design conditions

The modelling tests performed in the previous sections were also useful to verify the influence of approximating the pump efficiency to a constant efficiency. Nevertheless, the pump efficiency is not the only factor that influences the possible savings obtained from the use of VFD. Several authors have mentioned and discussed other factors such as the geometry (or shape) of the pump impeller and the pump specific speed which affect the characteristic curve of the pump (Quintela, 1981), or even factors related to the piping system such as the system static head Marchi, Simpson, and Ertugrul (2012); Morton (1975); Quintela (1981) or the system friction losses (Morton, 1975), which, in turn, affect the system head curve.

It is known that the existence of a static head in the system curve limits the minimum speed rate allowed for the pump since the pump shut-off head can never be inferior to the system static head. At the same time, the shape of the system head curve, which is also influenced by the friction head losses, can cause the pump curve intersection with the system curve in regions of low efficiency. Friction head losses (considered as a major loss) are dependent on the pipes surface material, which increases the viscous effects, and hence the energy loss, with the increase of the surface roughness. The friction losses also include losses due to obstructions in pipes (usually called minor losses). The losses related to the roughness of the pipe surface tend to increase with the age of the pipe. Nevertheless, the influence of changing some parameters in a system head curve is not often quantified in scientific works.

In this section, the influence of using a variable-speed pump and also of allowing distinct pump operating time-steps is tested under two distinct changes in the pipe systems: (i) the static head loss and (ii) the friction head losses by changing the pipes roughness. Both tests are performed using the developed Excel-based tool in order to compare results of both energy consumption and costs.

12.4.1 Changing the geometric head

In order to verify the differences in savings resulted from the use of variable pump speeds and operating times in similar systems but presenting distinct geometric heads (static heads), the initial case-study network, with a static head of 90 m, is used as reference. The same system is tested considering an inferior static head of 30 m and a superior static head of 120 m.

The pump controls and the tank water levels variation resulted from such controls for the initially considered network were already provided in figures 12.3 and 12.9. Respecting the systems with distinct geometric heads, the 24-hour pump controls were adjusted to each network characteristics in order to not violating the system constraints such as the maximum/minimum tank water levels allowed as well as keeping the same tank water level in the begin and in the end of the represented day and even constraints related to the nodal pressures (which cannot present negative values). The pump controls considered for the systems with a smaller and a higher geometric head, are represented in figures 12.11 and 12.12, respectively. Figure 12.10 also provides the resulting pump operating points for the two type of controls considered. It is possible to verify that, in the system with the larger geometric head, the speed is never reduced to 80 % of the nominal, since the head provided by the pump (see the the pump characteristic curve at $0.8N$) is very close to the minimum head required by the system (120 m).

As can be observed in figures 12.11a and 12.12a, the use of fixed 1-hour time-steps for the pump

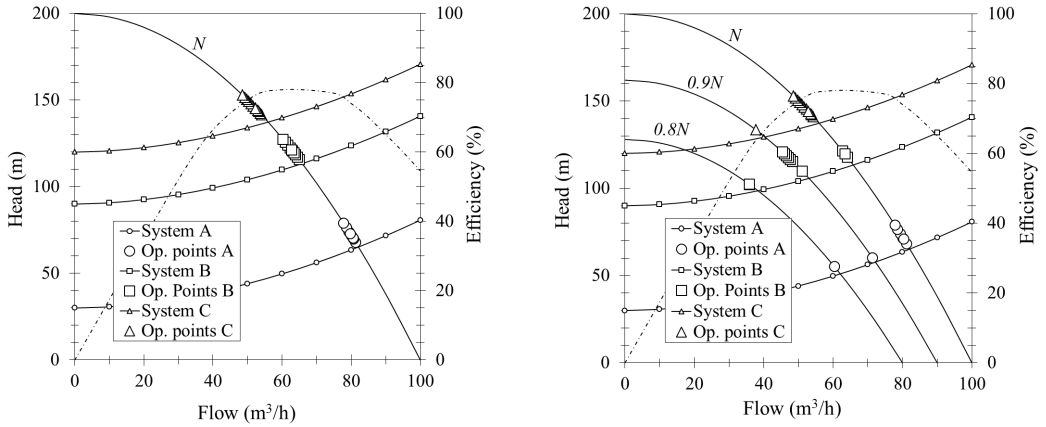


Figure 12.10: Results of the pump operating points, considering (a) a fixed-speed pump and (b) a variable-speed pump, for three system head loss curves with distinct static heads: (i) system A, with a static head of 30 m, (ii) system B, with a static head of 90 m and (iii) system C, with a static head of 120 m.

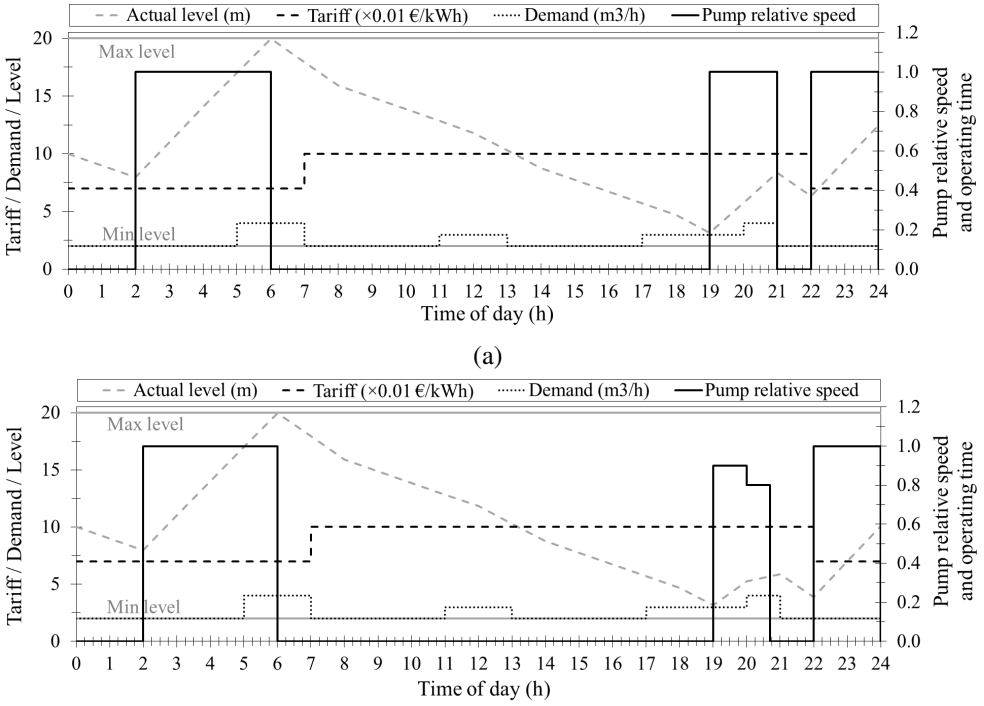
operation does not allow reaching a final water level equal to the initial. Instead, the final water level in tank is always superior, meaning that more water than the needed is being pumped. Making small adjustments in the pump operating time makes possible to pump only the water needed to satisfy the imposed constraints and, at the same time, allows to reduce the volume of water pumped, implying a reduction in the total energy consumption and cost. The quantified reductions are provided in table 12.3.

Table 12.3: Results obtained from varying the pump speed and operating time in the system of the case-study considering distinct values for the static head: (i) 90 m, corresponding to the initial system, (ii) an inferior value of 30 m, and (iii) a higher value of 120 m.

Type of control	Static head (m)	Avg Eff. (%)	Daily cost (€)	Avg energy (kWh/m^3)	Pumped water (m^3)	Daily energy (kWh)
fixed	30	74.07	13.18	0.21	638.13	170.57
speed	90	77.82	21.01	0.42	627.69	266.14
and time	120	75.35	25.44	0.53	619.58	328.26
variables	30	75.06	11.33	0.18	590.92	151.81
speed	90	71.78	19.54	0.44	590.48	257.07
and time	120	74.29	24.02	0.53	590.43	314.03

	Savings from pump speed and operating time variation (%)			
Static head	Cost	Avg energy	Water	Energy
30	14.09	5.52	7.40	11.00
90	6.99	-3.30	5.93	3.41
120	5.60	-0.81	4.71	4.34

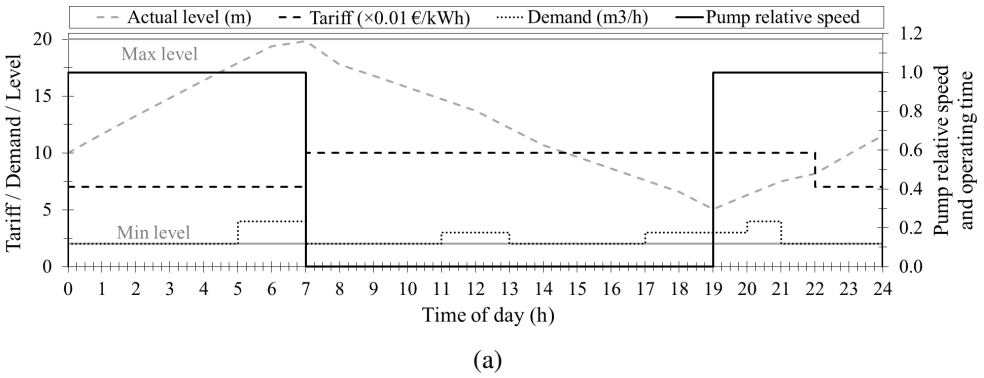
Observing the results for the percentage of savings resulted from varying the pump speed and operating times (table 12.3), it is possible to conclude that the pipe networks characterised by smaller geometric heads present greater chances of reducing the operational costs and associated energy consumption. This can be explained by the fact of such systems have more possibilities of taking advan-



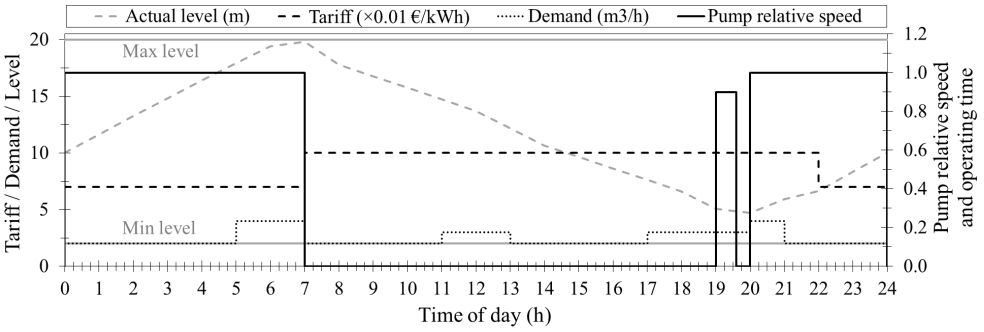
(a)

(b)

Figure 12.11: Results for the simulation of 24 hours of operation of the case-study network with a geometric head of 30 m considering: (a) a fixed-speed pump with fixed 1-hour operating time-steps and (b) a variable-speed pump with variable operating times.



(a)



(b)

Figure 12.12: Results for the simulation of 24 hours of operation of the case-study network with a geometric head of 120 m considering: (a) a fixed-speed pump with fixed 1-hour operating time-steps and (b) a variable-speed pump with variable operating times.

tage of the pumps speed variation, which allows an adaptation of the pumps to a large set of flow and head combinations.

In the system with a static head of 30 m (figure 12.11), a simple change in the pump operation in the time-step between the 20 and 21 hours (both speed and operating time reduction) allowed 14.09 % of cost reduction with a decrease of 11 % in the daily energy consumption. Such improvements are due to a reduction in both the volume of water pumped (reduction of 7.4 %) and the average energy consumed per unit of water pumped (reduction of 5.52 %).

Concerning the system with the largest static head (120 m), since the pump needs to overcome a greater difference in elevation, more energy is required for pumping the water to the tank. Indeed, this system presents the largest values for the daily energy consumption and cost. At the same time, it demonstrates to be the system with lower chances to improve its operation in terms of cost and energy since, due to the system curve, the pump is not allowed to reduce so much its rotational speed. Since the curve is more close to the initial system (90 m of static head), the savings obtained from reducing both the speed and operating time of the pump are closer to the ones obtained for the initial system. The initial system presented 3.41 % and 6.99 % of reduction in energy and cost, respectively, while the system with 120 m of static head reduced 4.34 % and 5.60 % its daily energy consumption and cost, respectively.

Results presented in table 12.3 also shows that, although presenting the larger energy savings, the system with the smaller static head also presents the lower value for the average efficiency when the pump is operating at fixed-speed. As can be observed in figure 12.10a, the system curve does not cross the pump curve at the best efficiency point, leading to an operation at a reduced efficiency. On the other side, the variation of the pump speed allows to move the pump operation to more efficient points, leading to the highest value of average efficiency (75.06 %) when compared to the other systems. This example demonstrates the benefit that the recourse to VFD can represent in real systems where frequently the pumps are dimensioned considering future requirements of the network, leading to the installation of oversized pumps considering the actual needs.

12.4.2 Changing the pipes roughness

Similarly to the analysis presented in section 12.4.1, in this section the case-study network provided initially is used as reference to analyse the influence of the pipe roughness coefficients, namely $C = 30$ and $C = 140$ (where $C = 50$ is the value considered initially). Lower values of the roughness coefficient are the equivalent to older pipes and usually the higher values correspond to the new ones. The head loss curves that characterise the analysed systems are provided in figure 12.13.

Figure 12.13 shows that lower values of roughness coefficients provide the steepest curves, which are related to the increase of the pipes head losses. It can be also observed that, although the lower coefficient ($C = 30$) is substantially closest to the initial value ($C = 50$) when compared to the largest coefficient ($C = 140$), such difference is not notorious in the graphical representation of the resulting head losses in the systems. This is explained by a fast decrease of the resistance coefficient with the increase of the Hazen-Williams roughness coefficient.

In this case, the pump operation had also to be adjusted to the distinct characteristics of the systems in order to satisfy water levels and pressure constraints. The pump controls initially considered for the systems with lower and higher pipe roughness coefficients are provided in figures 12.14a and

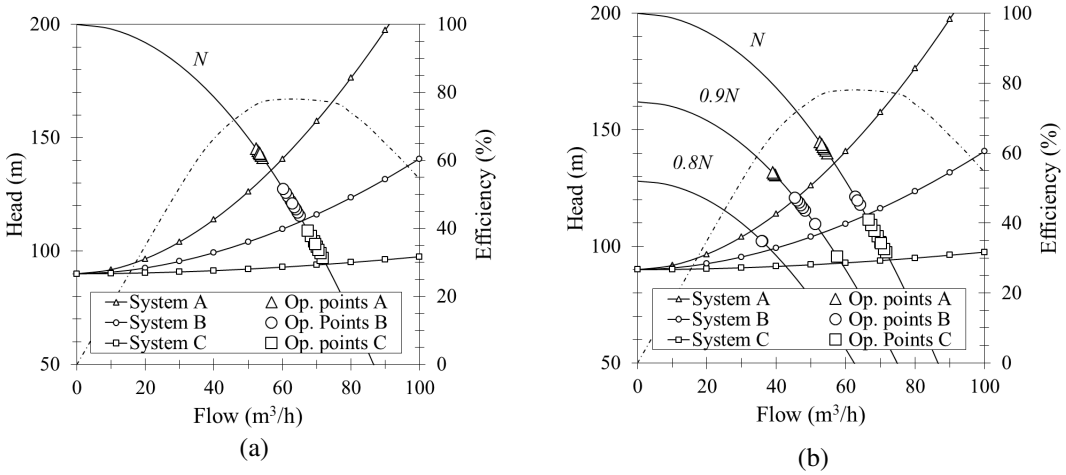


Figure 12.13: Pump operating points, considering (a) a fixed-speed pump and (b) a variable-speed pump, for three system head loss curves with distinct pipe roughness: (i) system A, with a roughness coefficient of 30, (ii) system B, the initial system, with a roughness coefficient of 50 and (iii) system C, with a roughness coefficient of 140.

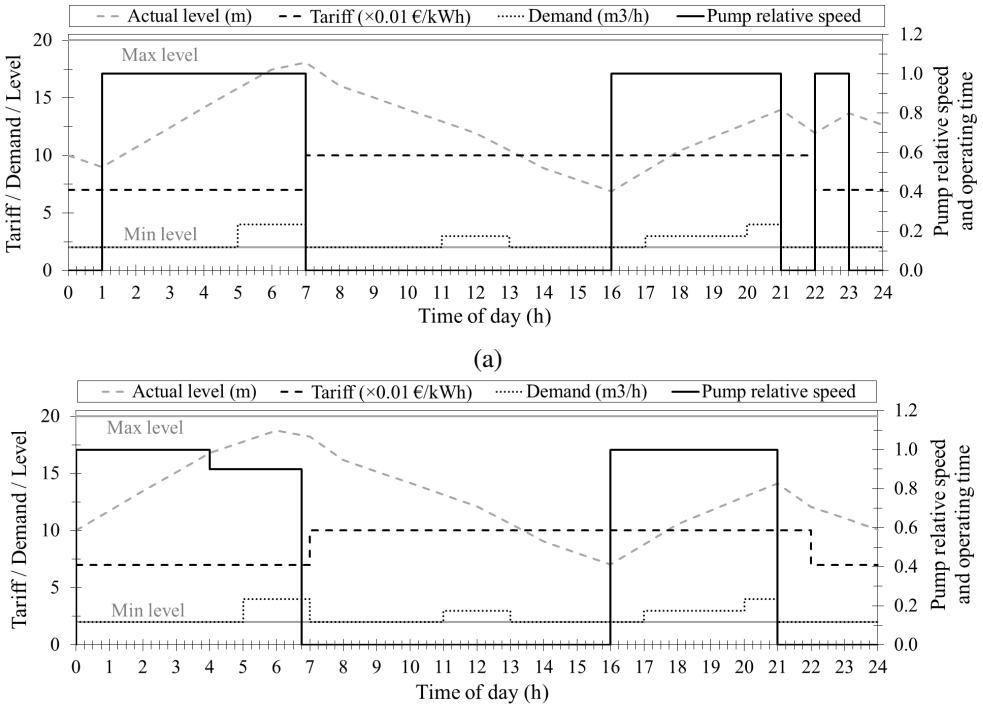
12.15a, respectively. Table 12.4 provides the results of energy, costs, efficiency and water pumped for each distinct system comparing the initial operation at fixed speed and also fixed 1-hour operating times with the improved operation considering variable operating times and speeds for the pump.

Table 12.4: Resume of the results obtained from varying the pump speed and operating time in the case-study network considering distinct values for the roughness coefficient: (i) the initial value considered, 50, (ii) a lower coefficient of 30, and (iii) a higher coefficient of 140.

Type of control	Rough. coeff.	Avg Eff. (%)	Daily cost (€)	Avg energy (kWh/m^3)	Pumped water (m^3)	Daily energy (kWh)
fixed	30	76.04	27.08	0.51	641.64	328.30
speed	50	77.82	21.01	0.42	627.69	266.14
and time	140	77.38	17.43	0.37	624.37	227.78
variable	30	73.16	25.54	0.52	590.99	306.28
speed	50	71.78	19.54	0.44	590.48	257.07
and time	140	77.37	16.19	0.36	591.81	215.25

Savings from pump speed and op. time variation (%)				
Rough. coeff.	Cost	Avg energy	Water	Energy
30	5.69	-1.87	7.89	6.71
50	6.99	-3.30	5.93	3.41
140	7.12	0.67	5.22	5.50

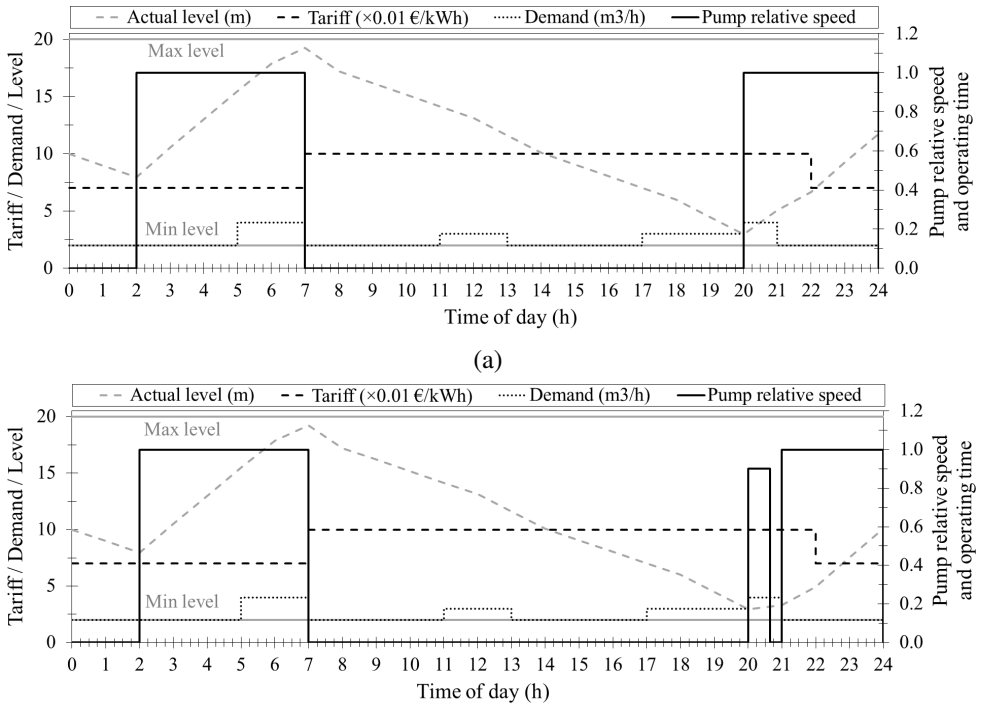
Before starting to analyse the results provided in table 12.4, it is important to provide some observations respecting the operation initially considered for the system with a roughness coefficient of 30 (figure 12.14a). Intuitively, an operator tends to shut the pump off during the high cost period but, in this situation, it is necessary to keep the pump on from 4 p.m. to 9 p.m.. Due to the larger pipe head losses in this system, the water level in tank cannot reach lower values than the represented. Otherwise, when the water demand increases between the 20 and 21 hours, the pressure in the demand node that is supplied by the tank reaches negative values, i.e., the system is not able to supply the con-



(a)

(b)

Figure 12.14: Results for the simulation of 24 hours of operation of the case-study network with a pipe roughness coefficient of 30 considering: (a) a fixed-speed pump with fixed 1-hour operating time-steps, and (b) a variable-speed pump with also variable operating times.



(a)

(b)

Figure 12.15: Results for the simulation of 24 hours of operation of the case-study network with a pipe roughness coefficient of 140 considering: (a) a fixed-speed pump with fixed 1-hour operating time-steps, and (b) a variable-speed pump with also variable operating times.

sumers while satisfying the pressure constraints. Even with this limitations in the system operation, some speed and operating time reductions allowed savings in the daily energy consumption and cost of 6.71 % and 5.69 %, respectively.

Results presented in table 12.4 demonstrate that the pipe roughness can be a factor with influence in the possible savings obtained from the use of VFD. Systems characterised by lower roughness coefficients (and hence, higher pipe roughness) resulted in less savings in the operational costs. On the other hand, the system with the lowest roughness coefficient ($C = 30$) presented the highest water and daily energy savings. This can be related to the higher initial values of water pumped and energy consumption due to the operational limitations resultant from the pressure constraints. Also due to these same pressure constraints, the pump operation was maintained during the high cost period which resulted in less cost savings.

Table 12.4 also shows that the system with $C = 50$ presented the lowest average efficiency when controlled with variable pump speed and variable operating times. This is a result of the pump operating at 80 % of its nominal speed, and hence at lower efficiency points, which does not occur in the other two systems (see the operating points at distinct speeds for the three systems in figure 12.13b).

It should be mentioned that, although the roughness coefficient of 50 has been chosen for the system initially considered in this work, values between $C = 100$ and $C = 140$ are usually stated for pipes in water distribution systems. For this reason, systems with roughness coefficients of 100 and 120 were also tested. However, no difference in savings from the use of variable speed were observed when compared to the system with $C = 140$.

12.5 Assessing the use of distinct formulae for the pumps efficiency computation

After the topic discussed in Chapter 12 on how to predict the pump efficiency with speed reduction, it comes the question of *"which of the discussed methods is the most adequate?"* or even *"what is the influence on the energy consumption and cost prediction of using distinct formulae for the efficiency computation?"*. At the same time, as already stated by Marchi and Simpson (2013), EPANET does not use correctly the affinity laws for the efficiency prediction with the reduction of the pump speed. Instead of considering the speed-adjusted curve for the efficiency prediction in case of speed reduction, EPANET only searches for the efficiency correspondent to the speed-adjusted flow by interpolation in the efficiency curve for the nominal speed.

Following the previously mentioned issues, the following section intends to provide a comparative study between distinct methods for the efficiency prediction, and hence, for power and energy computation.

12.5.1 Effect of using distinct formulae in energy savings computations

At a first stage, the theoretical speed-adjusted curves for both pump efficiency and power were represented considering four distinct methodologies: (i) the affinity laws (AL), (ii) the formula of Sárbu and Borza (1998) (SB), (iii) the method followed by EPANET (EPA), and (iv) the graph-based formula proposed in this work. Figure 12.16 provides such theoretical curves, considering the nominal speed, N, and also 60 %, 80 % and 120 % of the nominal speed ($0.6N$, $0.8N$ and $1.2N$, respectively)

for a pump with the characteristics of the pump represented in the simple network discussed in this chapter.

After the theoretical curves representation, it seemed relevant to quantify the savings possible to obtain from the use of variable speed drives when considering, for such savings computation, the distinct methods of efficiency prediction. Thus, the energetic and economic performance of the single-pump network, initially described in section 12.1, is tested again for a type of pump control at both fixed- (figure 12.3) and variable-speed (figure 12.5), considering the distinct methods for the pump efficiency computation. Both tests were also performed using the Excel-based tool specifically adapted for each distinct efficiency computation method. Results are provided in table 12.5.

Table 12.5: Results obtained by using four distinct methods to predict the pump efficiency with speed variation: (i) the Affinity Laws (AL), (ii) the formula proposed by Sárbu and Borza (1998) (SB), the method used by EPANET 2.0 (EPA) and the formula proposed in this work.

Type of control	Eff. method	Avg eff. (%)	Avg power (kW)	Daily cost (€)	Avg energy (kWh/m^3)	Pumped water (m^3)	Daily energy (kWh)
fixed speed	All	77.82	26.61	21.01	0.424	627.69	266.14
variable speed	AL	75.86	22.56	18.82	0.419	590.01	248.14
	SB	77.55	22.21	18.48	0.410	590.01	244.34
	EPA	73.25	23.21	19.47	0.436	590.01	255.34
	Proposed	77.57	22.20	18.47	0.410	590.01	244.25

	Savings from pump speed variation (%)				
Method	Avg power	Cost	Avg energy	Water	Energy
AL	15.24	10.40	1.23	6.00	6.76
SB	16.54	12.04	3.33	6.00	8.19
EPA	12.78	7.34	-2.66	6.00	4.06
Proposed	16.57	12.07	3.37	6.00	8.22

Respecting the efficiency prediction using the affinity laws, for each non-nominal pump speed, an adjusted efficiency curve was constructed. Then, using the value of the speed-adjusted flow, the speed-adjusted efficiency was obtained by interpolation in the corresponding curve.

The results for the method followed by EPANET correspond to the ones already provided in section 12.3.1. The efficiency for each new speed is also obtained by interpolating the speed-adjusted flow in the efficiency curve but, in this case, the same initial curve (correspondent to the nominal speed) is used (see figure 12.16).

For the methods applying the Sárbu and Borza equation as well as the equation proposed in this work, the efficiency at each time-step was determined according to the following equations previously presented in the first chapter of the Methodology and mathematical modelling Part (Chapter 7):

$$\eta_2 = 1 - (1 - \eta_1)\left(\frac{N_1}{N_2}\right)^{0.1}, \tag{12.1}$$

for the method proposed by Sárbu and Borza (1998), and

$$\frac{\eta_2}{\eta_1} = \left(\frac{N_2}{N_1} - 1\right)^3 + 1, \tag{12.2}$$

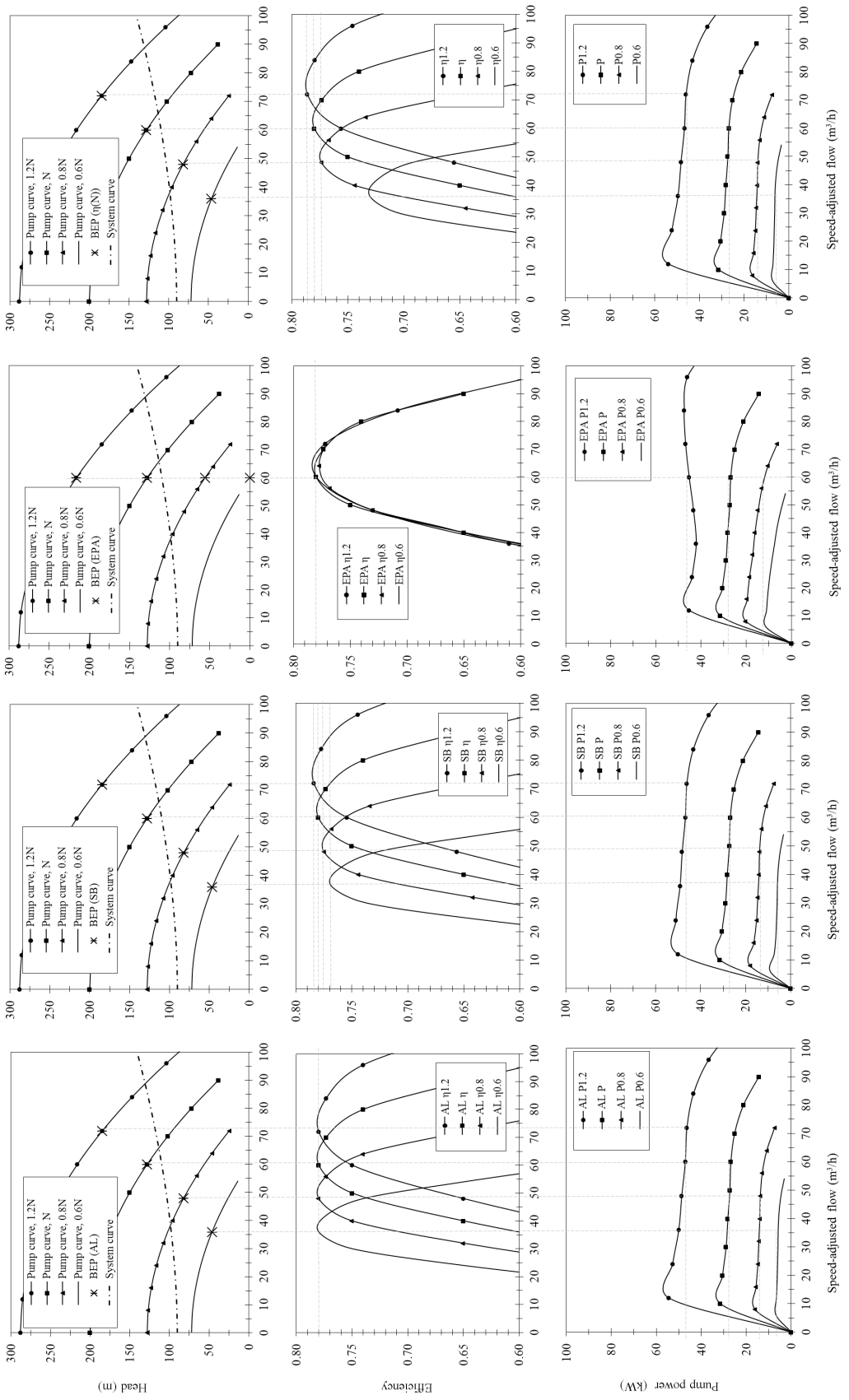


Figure 12.16: Pump performance curves at nominal and 60 %, 80 % and 120 % of the nominal speed for distinct efficiency prediction methods: (i) the affinity laws (AL), (ii) the formula proposed by Sárbu and Borza (1998) (SB), (iii) the EPANET method (EPA), and (iv) the formula proposed in this work.

for the method proposed in this work.

Observing the curves provided in figure 12.16, significant differences in the curves predicted for the distinct pump rotational speeds are presented. On the other hand, the power curves predicted by all the methods except EPANET, appear to be similar. The power curves predicted by EPANET, when compared to the other three methods, demonstrate an overestimation of the power consumption in certain flow regions and an underestimation in others, which, in turn, can lead to incorrect evaluations of possible savings, following what was stated by Marchi and Simpson (2013).

Results of table 12.5 confirm the previous statement respecting the EPANET pump efficiency prediction. Comparing with the other methods, EPANET clearly underestimates the savings obtained from the use of a variable-speed pump. This is explained by the lower average efficiency (73.25 %) when compared to the SB method (77.55 %), leading to higher power and energy consumptions and, consequently, to more costs and then, less savings. While EPANET predicted a percentage of savings in energy consumption and costs of 4.06 and 7.34, respectively, the other methods predicted savings in the range $6.76 - 8.22$ % for the energy consumption and $10.40 - 12.07$ % for the associated costs.

It must be remembered that, for a distinct case (with, for instance, a distinct pump or even a distinct system head curve), the results obtained by EPANET may overestimate the real savings.

The method based on the equation proposed in this work provided very similar results to the ones obtained by the Sárbu and Borza (1998) method. However, it should be noticed that the operational conditions meet the region where this method approximates the later, *i.e.* the pump speed is never below 80 % of the nominal speed and the original pump efficiency (η_1) is superior to 0.7.

As can be observed by the results obtained for the method following the affinity laws, if EPANET applied correctly the affinity laws in its efficiency prediction, the computed savings would be closer to the ones obtained by the SB method. On the other hand, it should be also noted that this is a conclusion for this specific case-study, since the affinity laws fail in the prediction of the pumps behaviour under certain conditions (such as lower pump speeds and dimensions).

The studies provided in this section do not allow to answer the first question made in the beginning of the section ("*which of the discussed methods is the most adequate?*"). Only additional experimental tests can provide a general answer to such question. However, under the conditions of the tested simple network, the proposed formula can be set at a similar level of accuracy of the SB formula. Both formulas demonstrate to be significantly more accurate than the so commonly used EPANET.

There is a clear necessity of real data, provided by the pumps manufactures, for instance, in order to obtain more conclusive results concerning the best possible alternatives for the replacement of the less-accurate methods commonly used in hydraulic simulators. This lack of information has also been mentioned by researchers dealing with efficiency studies concerning the use of pumps as turbines (PAT) for energy recovery in WSS (Carravetta, Fecarotta, Martino, & Antipodi, 2014).

Respecting the second question, the effects of using distinct efficiency prediction methods demonstrate to be considerable since results provided appreciable differences in the possible savings from the use of variable-speed drives.

12.5.2 Formulations comparison with existing experimental data

Simpson and Marchi (2013) provided experimental data for the best efficiency point (BEP) of pumps with distinct dimensions (small to large) operating at distinct speeds. Such data was compared with

the values predicted by the affinity laws and by the approach of Sárbu and Borza (1998). As expected, for the large pump, the experimental values of the efficiency change were not significant and both theoretical approaches provided satisfactory predictions. However, for smaller pumps, the BEP was significantly reduced with the speed decrease. While the affinity laws (AL) predict that the BEP is maintained the same, the SB equation was able to estimate speed-adjusted efficiency values near the experimental data.

T. Walski, Zimmerman, Dudinyak, and Dileepkumar (2003) also performed an experiment to determine how well the affinity laws apply to a variable-speed pump. The experimental data agreed with the pump head curves generated with the affinity laws. However, concerning the pump efficiency curves, the shape of the experimental results agreed with the curves predicted by the affinity laws but the measured values were considerably inferior to the predicted ones. Moreover, it was observed an increase in these deviations with the speed decrease. T. Walski et al. (2003) also pointed out the decrease of the VFD efficiency with the speed reduction as the main factor for the overall efficiency bellow the expected.

Nevertheless, it should be noticed that the experiment presented by T. Walski et al. (2003) was performed with a very small pump (0.37 kW of power, nominal flow of 0.63 l/s and nominal head of 0.6 m), which leads to expect greater reductions in the efficiency with the speed decrease and does not allow to make general conclusions on the influence of variable-speed in the pumps efficiency curves.

Following the same approach generally applied in this field, and in order to provide more conclusive results concerning the formulation proposed in this work, experimental data found in the literature was used to evaluate the performance of the proposed equation for the speed-adjusted efficiency curves (Coelho & Andrade-Campos, 2016a).

Table 12.6 lists experimental data for the best efficiency point (BEP) of two distinct pumps: one of large dimension (556 kW) and one of small dimension (5.5 kW). Such data were obtained from the works of Ulanicki, Kahler, and Coulbeck (2008) and Simpson and Marchi (2013) and were already used by the last ones to compare the performance of the affinity laws with the formulation proposed by Sárbu and Borza (1998). Results of the BEP with speed reduction using the distinct approaches discussed in this work (including the Sarbu and Borza formulation - SB -, the Affinity Laws - AL -, the EPANET approach - EPA -, and the proposed formulation - Prop) are also provided.

Table 12.6: Experimental and predicted values for the best efficiency point (BEP) of two real pumps. Prediction results obtained using distinct methods: (i) the Affinity Laws (AL), (ii) the formula proposed by Sárbu and Borza (1998) (SB), the method used by EPANET 2.0 (EPA) and the formula proposed in this work (Prop).

		Experimental		Prediction				Difference (%)			
Pump type	Power (kW)	BEP at N_1 (η_1)	BEP at N_2 (η_2)	BEP AL (η_2)	BEP SB (η_2)	BEP EPA (η_2)	BEP Prop (η_2)	AL	SB	EPA	Prop
Large[(1)]	556	83.60	83.50	83.60	83.18	83.60	82.65	0.12	-0.39	0.12	-1.02
Small[(2)]	5.5	56.00	52.00	56.00	53.34	56.00	51.08	7.69	2.57	7.69	-1.76

[(1)] Sulzer pump HPL 54-30-20; data reported by Ulanicki et al. (2008). N_1 = 1525 rpm, N_2 = 1182 rpm;
[(2)] Pump 50-32-160 HT, from TKL catalogue (1989); data reported by Simpson and Marchi (2013). N_1 = 3600 rpm, N_2 = 2000 rpm.

For the large pump, when the speed is reduced from 1525 to 1182 rpm, the most reliable results

seems to be obtained using the Sárbu and Borza (1998) methodology. However, for the small pump, where the efficiency changes are usually more significant and difficult to predict, when the speed is reduced from 3600 to 2000 rpm, the formula proposed in this work can predict the efficiency with a lower error than the other methods.

Nevertheless, Table 12.6 only compares the BEP values, considering that pumps are always operating in the same point. However, this type of comparison between methods is not sufficient for pumps expected to operate at different speeds and flow rates, where more efficiency points should be evaluated.

A comparison between the distinct formulations considering multiple points of an experimental efficiency curve is depicted in Figure 12.17. Such curve is representative of the large Sulzer pump (HPL 54-30-20), where the proposed formula present the largest absolute difference in the BEP. As can be observed, the predicted efficiency curve fit quite well the experimental curve retrieved from Ulanicki et al. (2008). The average fitting error for the proposed formulation is 1.2 %. However, the efficiency curve predicted by the EPANET methodology (labelled as η_1), whose difference in BEP was only 0.12 %, cannot accurately predict the experimental values for η_2. The Sárbu and Borza (1998) formulation can also fit well the experimental results, however with an average error of 2.1 %.

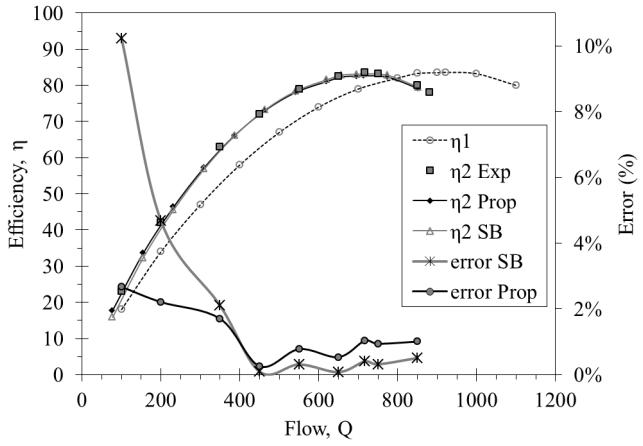


Figure 12.17: Comparison between an experimental efficiency curve (data retrieved from Ulanicki et al., 2008) and the predicted curves using distinct methods. The method used in the EPANET is defined as η_1.

Figure 12.17 demonstrates that the proposed formula can accurately predict pump efficiency curves. However, further comparisons with the curves of other types of pumps and of different sizes should be performed, for testing (a) the proposed equation and (b) all the previously discussed approaches.

13. Optimising the operations of water supply networks

The operational control optimisation tool developed in this thesis is used to minimise the daily energy costs in five different models of water supply systems, including two simplified models of real networks. Distinct optimisation techniques are tested in selected networks and, additionally, a sensitivity analysis to the numerical parameters of some algorithms is also performed.

13.1 Networks description and initial conditions

Five water supply networks were selected to test the developed tool in problems consisting not only in different number of pumps and valves, and hence, different type and number of decision variables, but also presenting different number of storage tanks, pipes, sources and different demands, which provide different levels of complexity to the networks, and therefore, to the optimisation algorithms. Table 13.1 summarises the main characteristics of the five tested networks.

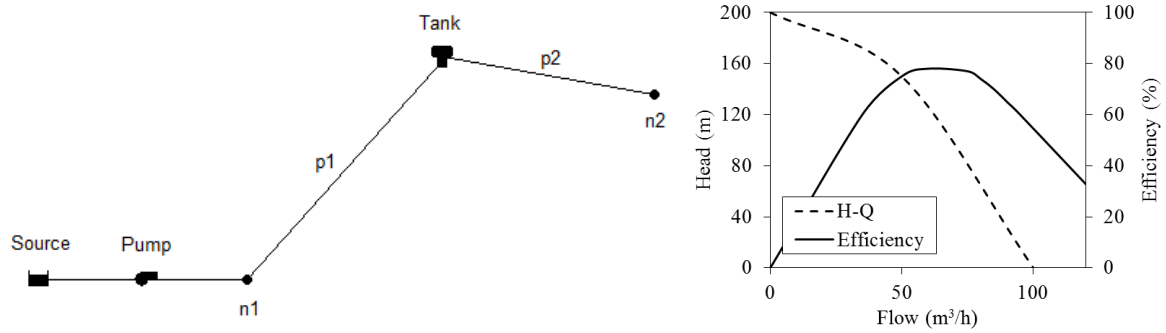
Table 13.1: Overview of the main characteristics of the networks used for optimisation.

Tested network	VS pumps	FS pumps	Valves	Decision variables	Storage tanks	Pipes	demand nodes	Water sources
Single-pump	1	_	_	48	1	2	1	1
Van Zyl (a)	2	1	_	120	2	15	2	1
Van Zyl (b)	3	_	_	144	2	15	2	1
Walski	2	_	2	96	1	20	8 (1 fire flow)	2
Richmond	7	_	_	336	6	44	11 (1 negative)	2
Portuguese	4	_	1	216	4	42	11 (1 negative)	_

The simulation period considered for the five networks was the traditional one day divided into hourly time-steps. This is the period typically used by other researchers in this type of problem, and hence, is the most adequate choice for comparative analysis. Additional costs due to peak power demand are not considered since this type of cost is commonly computed over a billing period (*e.g.* monthly). The method selected for computing the friction head losses in pipes was the Hazzen-Williams since this is the most common method in research works in the field.

13.1.1 Single-pump network

The simplest problem tested (Figure 13.1) corresponds to a network with the same configuration of the network used in Chapter 12, a single-pump network, with one reservoir and one tank responsible for supplying one point with variable demand. The input file of the network with the considered initial conditions is available in Appendix D.



(a) (b)

Figure 13.1: (a) EPANET model of the single-pump network and (b) the corresponding curves of the represented pump.

This problem represents the most simple case of a pumped water supply network, consisting in 48 decision variables (considering variable speed).

The initial operational conditions (initial solution) considered for this problem are provided in Figure 13.2. The pump is operating at nominal speed from 1h00 to 6h00 and from 19h00 (7 p.m.) to 24h00, remaining off during the rest of the day. Considering such conditions and the defined energy tariff, the associated daily pumping energy cost is 22.73 €, corresponding to a consumption of 26.44 kW (Table 13.2).

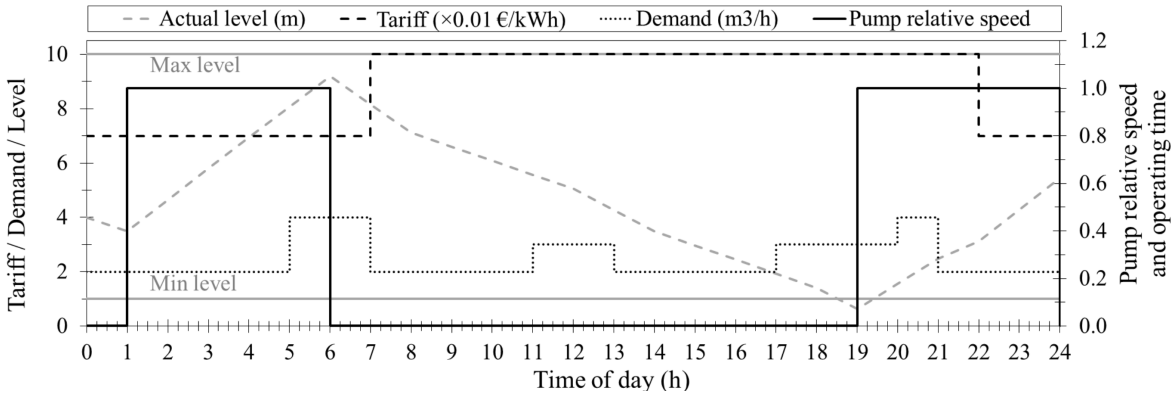


Figure 13.2: Main initial operational conditions of the single-pump network for a 24-hours period.

13.1.2 Van Zyl network

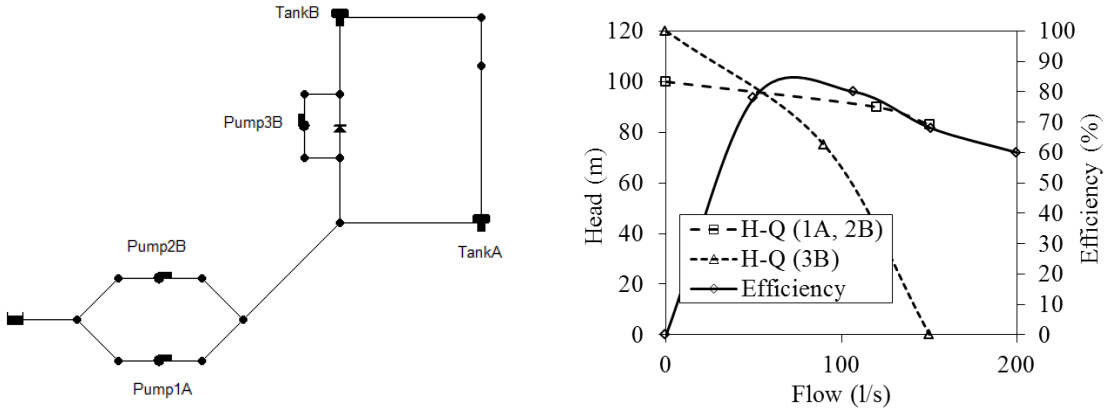
The van Zyl network is a test network that was originally proposed by van Zyl, Savic, and Walters (2004). Since then, the network was used by several authors for testing the control optimisation of fixed-speed pumps (López-Ibáñez, Prasad, & Paechter, 2011) and, more recently, considering also

Table 13.2: Results of the 24-hour simulation of the single-pump network considering the initial conditions.

Pump utilisation (%)	Avg Eff. (%)	Energy (kWh/m^3)	Avg power (kW)	Max power (kW)	Daily cost (€)
45.83	77.72	0.41	26.44	26.60	22.73
Total daily cost, $C_{\text{total}}(\mathbf{X}_0)$					22.73

variable-speed pumps (Coelho & Andrade-Campos, 2014a, 2014b; Hashemi, Tabesh, & Ataeekia, 2014).

The van Zyl network, represented in Figure 13.3, is composed of two storage tanks, A and B, with a zone of water consumption between them represented by two junction nodes. A pumping station composed of two pumps (1A and 2B) is responsible for pumping the water from the source to the tanks, with the support of a booster station containing one pump (3B).



(a) (b)

Figure 13.3: (a) EPANET model of the test network proposed by van Zyl et al. (2004) and (b) the corresponding curves of the represented pumps at nominal speed.

Two different optimisation possibilities are considered for this problem: (a) two variable-speed pumps (1A and 2B) and one fixed-speed pump (3B), which corresponds to 120 decision variables, and (b) three variable-speed pumps, corresponding to 144 decision variables (Table 13.1). The pumps characteristic cuves (H-Q) and the efficiency curve considered for the 1A and 2B pumps are represented in Figure 13.3b. For pump 3B, a constant efficiency of 85 % is considered.

Initial operational conditions

The main initial operational conditions as well as the energy tariff are represented in Figure 13.4. This initial solution is based on one of the optimal operational solutions proposed by van Zyl et al. (2004), with a daily energy cost of £ 345.24*. The simulation results in terms of energy and pumps utilisation considering such initial conditions are presented in Table 13.3. The corresponding EPANET input file that provides such results can be found in Appendix D. It should be noticed that, to consider the

*In this case, the same units considered by van Zyl et al. (2004) are used in order to facilitate the comparison of results.

pump 3B as a fixed-speed pump in the developed tool, a speed pattern cannot be associated to the pump and the time control statements should be changed to OPEN/CLOSED status.

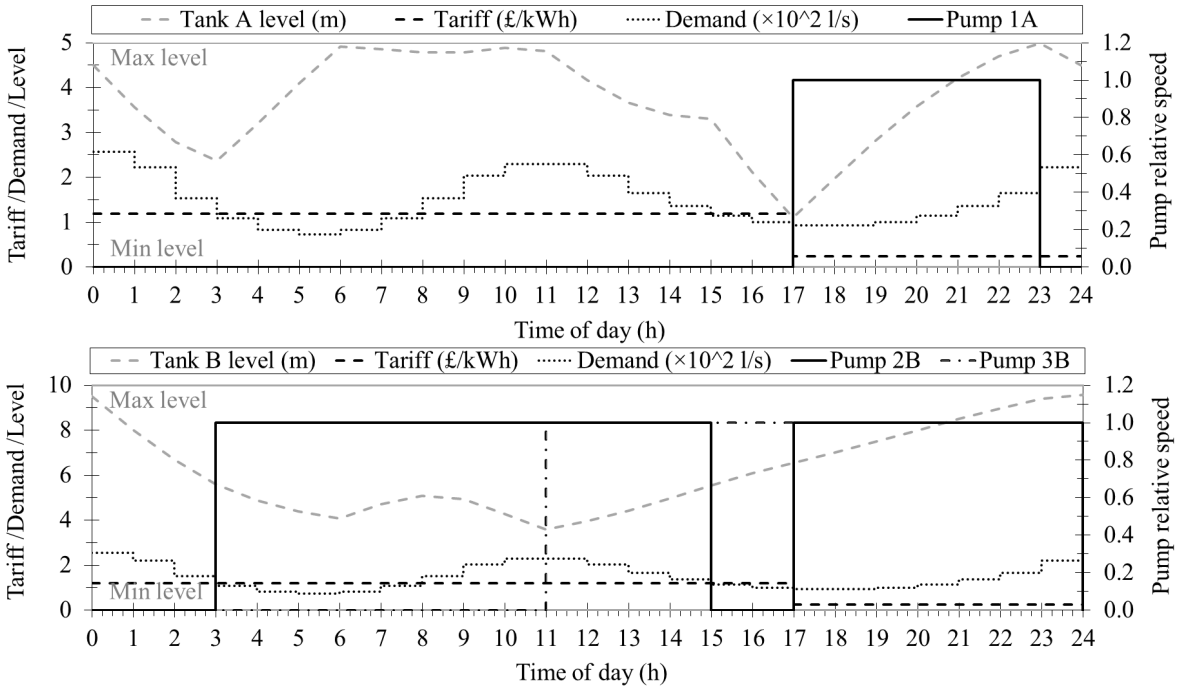


Figure 13.4: Main initial operational conditions of the van Zyl network for a 24-hours simulation period.

Table 13.3: Results of the 24-hour simulation of the van Zyl network considering the initial conditions.

Pump	Utilisation (%)	Avg Eff. (%)	Energy (kWh/m^3)	Avg power (kW)	Max power (kW)	Daily cost (£)
1A	25.00	75.48	0.32	143.09	147.03	20.95
2B	79.17	69.72	0.32	172.78	202.87	291.38
3B	54.17	85.00	0.07	35.58	47.28	32.91
Total daily cost, $C_{\text{total}}(\mathbf{X}_0)$						345.24

An important characteristic of the van Zyl network is concerned to the hours of the day in which the simulation occurs. The time presented in Figure 13.4 corresponds to the simulation time. However, it should be noticed that the clock-time for this specific problem is not coincident with the simulation time but, instead, starts at 7 a.m., meaning that the represented hour 0 corresponds to 7 a.m., the hour 1 corresponds to 8 a.m., and so on.

Impact of distinct efficiency formulae on savings computation

The discussion on the effect of using distinct formulae for the computation of the pumps efficiency with speed variation presented in Section 12.5 led to the conclusion that, for the single-pump network, the savings, both in terms of energy and costs, computed by EPANET underestimates the

savings computed with the use of other formulae for predicting the pump efficiency at distinct speeds. This same analysis on the savings computation was performed for the van Zyl network (Coelho & Andrade-Campos, 2016a). The daily costs for the network at two distinct operational conditions, *i.e.* with pumps running at fixed and at variable speed[†], considering the four methods previously stated for the speed-adjusted efficiency computation, are listed in Table 13.4.

Table 13.4: Savings results for the van Zyl network considering distinct pump efficiency prediction methods: (i) the Affinity Laws (AL), (ii) the approach proposed by Sárbu and Borza (1998) (SB), (iii) the method used by EPANET 2.0 (EPA) and (iv) the method proposed in this work.

Network	Type of control	Eff. curves	Daily cost (£)	Savings (%)
Van Zyl	Fixed-speed	Nominal	345.24	–
	Variable speed	AL	222.69	35.50
		SB	222.23	35.63
		EPA	231.16	33.04
		Proposed	221.27	35.91

In the van Zyl network, the daily energy cost computed by EPANET 2.0 also overestimates the values obtained with different speed-adjusted efficiency curves, leading to an underestimation of the savings computation (33 % of savings compared to the 36 % obtained with the use of other methods).

13.1.3 Richmond network

The Richmond network, represented in Figure 13.5, corresponds to a simplified model of the real Richmond network, part of the Yorkshire water supply area in the United Kingdom. This network is a benchmark of the Centre for Water Systems Resources of the University of Exeter, whose hydraulic model (EPANET input file) is available in the University website (University of Exeter, 2008). In such available EPANET model, the daily operational cost corresponds to £12316.79[‡]. In order to start the optimisation process with an initial solution that do not violate the constraints, the operation of the Richmond network is adjusted to prevent the water levels in the end of the simulation of being inferior to the levels in the beginning. Therefore, the operating costs were changed, leading to an initial solution of £15632.83 instead of £12316.79. Similarly with the van Zyl network, despite the shown simulation results begins at time zero, the clock-time of this network starts at 7 a.m..

The Richmond network is characterised by seven pumps (1A to 7F) and eleven demand nodes (J1 to J11). The efficiency and characteristic curves of all pumps are represented in Figure 13.6. The tariff associated to each pump as well as the demand pattern related to the demand junction nodes are presented in Figure 13.7. Node J3 has a negative demand associated, meaning that this is a zone of water inlet in the network.

[†]The solution for the pumps operating at variable-speed was obtained with the developed optimisation tool. One solution was randomly selected and, for such controls, the daily costs were computed according to each distinct method of efficiency computation.

[‡]In this problem, the provided units were also maintained.

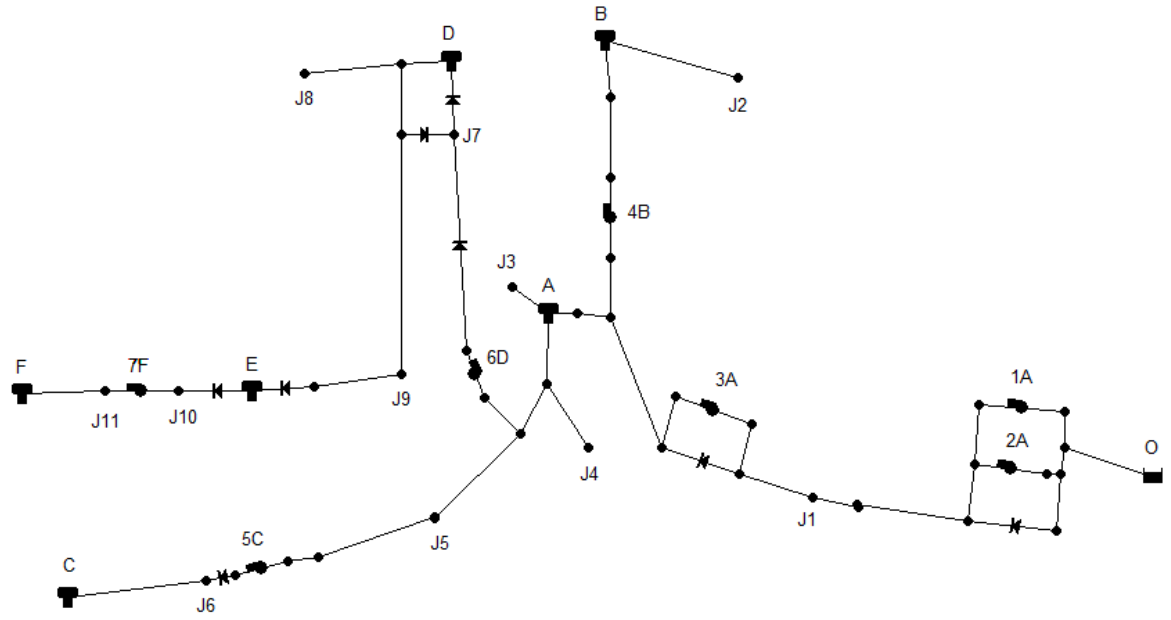
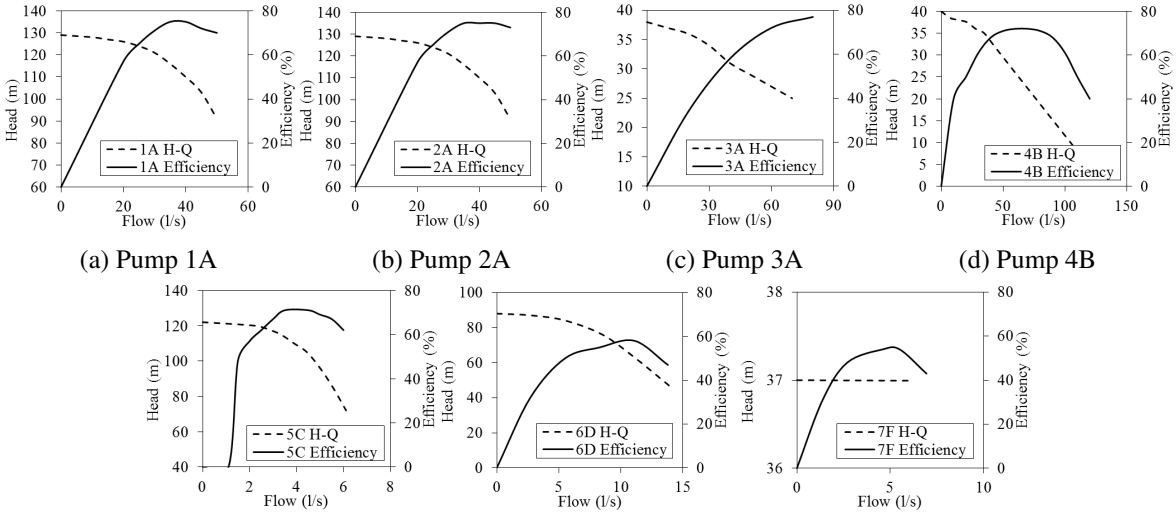


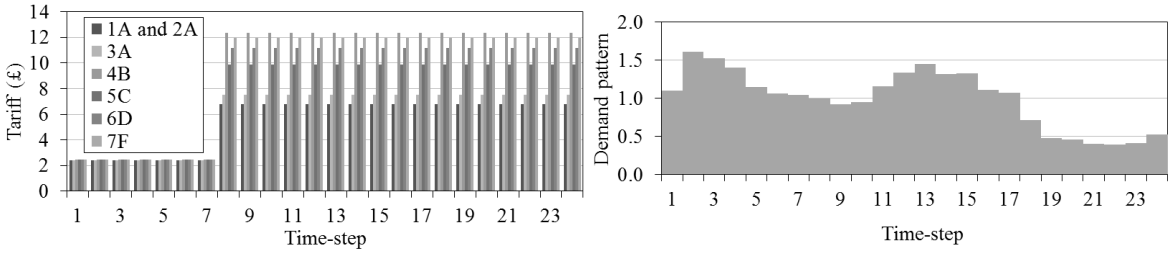
Figure 13.5: EPANET model of the Richmond network (University of Exeter, 2008).



(a) Pump 1A (b) Pump 2A (c) Pump 3A (d) Pump 4B

(e) Pump 5C (f) Pump 6D (g) Pump 7F

Figure 13.6: Efficiency and characteristic curves of the pumps represented in the Richmond EPANET model.



(a) (b)

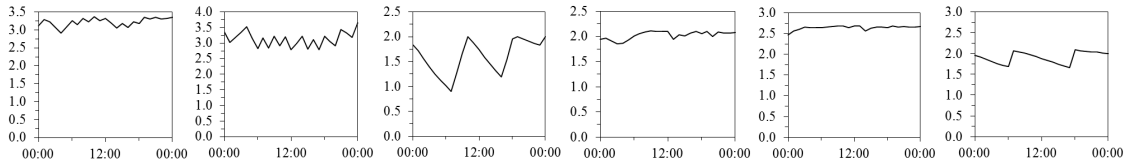
Figure 13.7: (a) Tariffs associated to each pump of the Richmond model and (b) water demand pattern associated to the demand junction nodes.

Initial operational conditions

The results in terms of pumps usage and efficiency for the 24-hours simulation of the network under the considered initial conditions are presented in Table 13.5. The results of such initial conditions are also translated in terms of water levels variation in each tank of the network in Figure 13.8. As can be observed, the water levels never reach the minimum levels of the tanks (0.0 m for all tanks) neither reach levels inferior to the initial ones in the end of the simulation.

Table 13.5: Results for the 24-hour simulation of the Richmond network considering the initial conditions.

Pump	Utilisation (%)	Avg. Efficiency (%)	Energy (kWh/m_3)	Avg. Power (kW)	Peak power (kW)	Daily cost (£)
1A	4.17	71.57	0.46	50.6	50.6	121.9
2A	100	65.32	1.34	57.74	60.61	7603.65
3A	100	51.05	1.54	18.23	22.05	2536.51
4B	50	63.42	0.16	18.2	18.5	1977.21
5C	29.17	54.4	8.02	19.53	112.82	1348.89
6D	95.83	46.93	0.66	10.67	11.86	2021.46
7F	8.33	27.92	0.36	1.61	1.61	23.22
Total daily cost, $C_{\text{total}}(\mathbf{X}_0)$						15632.83



(a) Tank A (b) Tank B (c) Tank C (d) Tank D (e) Tank E (f) Tank F

Figure 13.8: Variation of the water levels (m) in the tanks of the Richmond network during the 24-hours of simulation considering the initial operational conditions. The minimum and maximum levels of each tank are represented by the minimum and maximum values of the vertical axis of the corresponding chart.

Further details concerning the Richmond network initial operating conditions can be seen in the input model provided in Appendix D.

Impact of distinct efficiency formulae on savings computation

The same assessment of the impact of using distinct formulae for the speed-adjusted efficiency prediction on the computation of the energy savings in the Richmond network was performed (Coelho & Andrade-Campos, 2016a). The daily costs for this network at two distinct operational conditions, *i.e.* with pumps running at fixed and at variable speed[§], considering the four methods previously stated for the speed-adjusted efficiency computation, are listed in Table 13.6.

In the Richmond network, the differences between the savings computation considering distinct formulae are not so notorious. This can be mainly explained by the fact that the solution obtained for

[§]Similarly with the presented for the van Zyl network, a randomly selected optimisation solution for the Richmond network operating at variable-speed was used.

Table 13.6: Savings results for the Richmond network considering distinct pump efficiency prediction methods: (i) the Affinity Laws (AL), (ii) the approach proposed by Sárbu and Borza (1998) (SB), (iii) the method used by EPANET 2.0 (EPA) and (iv) the method proposed in this work.

Network	Type of control	Eff. curves	Daily cost (£)	Savings (%)
Richmond	Fixed-speed	Nominal	15632.72	–
	Variable speed	AL	12308.10	21.27
		SB	12310.10	21.25
		EPA	12403.06	20.66
		Proposed	12320.30	21.19

the network considering variable-speed pumps does not contain speed settings significantly different from the nominal speed, and hence, the resulting efficiency values are near the efficiency at nominal speed. However, the daily energy cost computed by EPANET 2.0 also overestimated the values obtained with the other methods for the efficiency prediction, leading to a slightly underestimation of the savings computation (less than 1 %).

13.1.4 Walski network

The Walski network, represented in Figure 13.9, was retrieved from the book of T. M. Walski et al. (2003), Chapter 10 - Operations (exercises section). This operations test network contains two water sources, multiple loops, valves and considers a fire occurrence in a certain period of the day, which represent significant differences from the previously presented water networks. The developed EPANET model according to the data available in T. M. Walski et al. (2003) is provided in Appendix D. The British units were converted to S.I. units. The same tariff and pump efficiency curve used in the van Zyl network were associated to the two pumps of this network since such information was not available in the original problem.

The Walski network has eight demand nodes (J3 to J6 and J8 to J11, in Figure 13.9), where J5 is the juntion node with the associated demand pattern that simulates a 3-hour fire occurrence from 11h00 to 14h00 (an additional demand of 42.75 l/s in each hour is considered).

The pumps operation, the tank water level variation and the considered demand pattern without fire flow for the 24 hours of simulation are presented in Figure 13.10. The simulation results in terms of pumps usage and efficiency are summarised in Table 13.7. Considering the presented operational conditions, this initial solution for the Walski network corresponds to a daily pumping energy cost of 324.36 €.

Table 13.7: Results for the 24-hour simulation of the Walski network considering the initial operational conditions.

Pump	Utilisation (%)	Avg. Efficiency (%)	Energy (kWh/m^3)	Avg. Power (kW)	Peak power (kW)	Daily cost (€)
PMP-1	58.33	76.29	0.54	187.78	415.17	237.22
PMP-2	25.00	75.76	0.33	121.64	284.60	87.14
Total daily cost, $C_{\text{total}}(\mathbf{X}_0)$						324.36

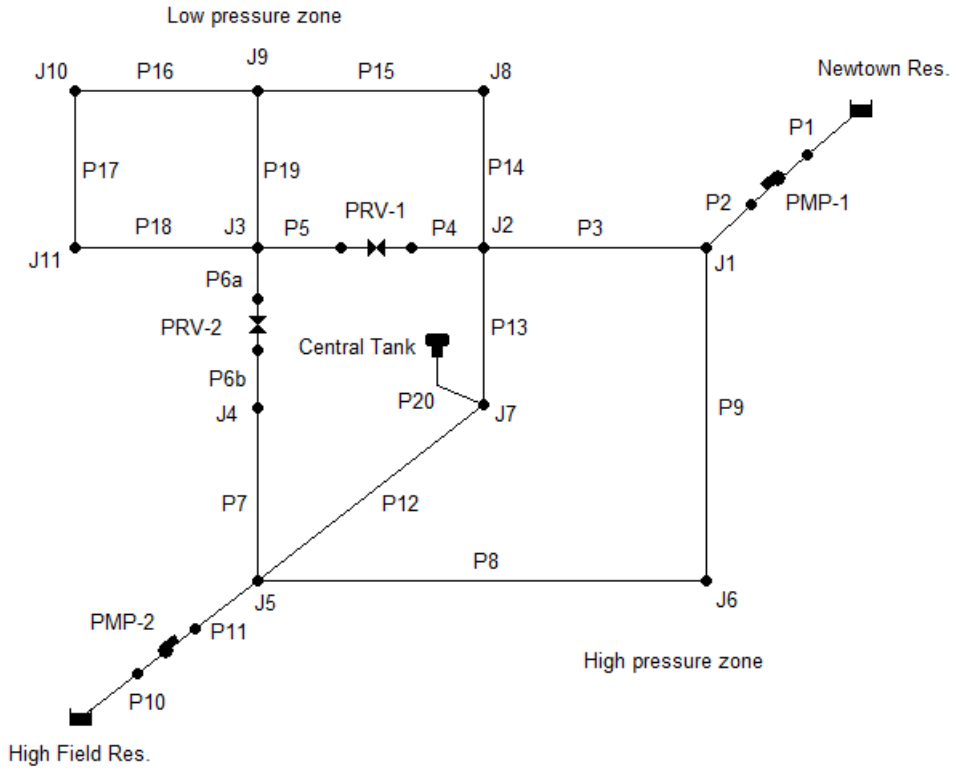


Figure 13.9: EPANET model of the Walski network. Data obtained from T. M. Walski et al. (2003).

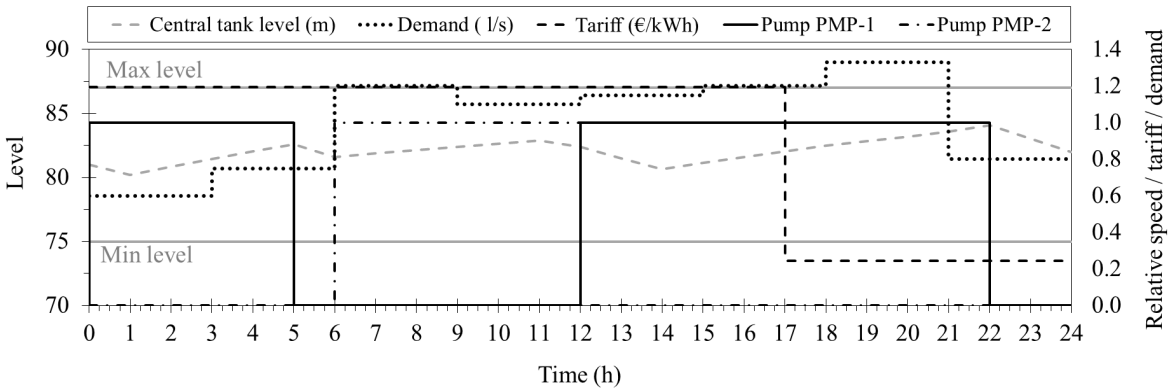


Figure 13.10: Main initial operational conditions of the Walski network for a 24-hour simulation period.

From Table 13.7, it can be observed that, contrarily to the observed for the previous networks, there is a significant difference between the average power consumed by the pumps during the 24 hours and the maximum power (peak power) consumed in that period, which is caused by the occurrence of a fire event.

13.1.5 Portuguese network

The Portuguese case-study addressed in this thesis corresponds to a part of a subsystem of a Portuguese multimunicipal water supply system, currently part of Águas do Norte Group. The entire system is responsible for the supply of more than 240 Mm^3 of water per day. The development of the network hydraulic model and its calibration were performed within a 1-year collaboration project (in 2013) with a Portuguese SME[¶]. CAD files containing the network main characteristics as well as measured real data used to perform the calibration were provided by the water utility. The simplified EPANET model of the Portuguese network is represented in Figure 13.11. The network consists in (i) four storage tanks (A to D), (ii) eleven demand nodes (D1 to D11), from which D1 has a negative base demand, representing the water inlet in the network (a source), (iii) two pumping stations (PMP-AB and PMP-CD) with two similar pumps each, and (iv) multiple valves (however, only one valve, G, is considered for control optimisation). This optimisation problem is then composed by 216 decision variables (4 variable-speed pumps and 1 valve).

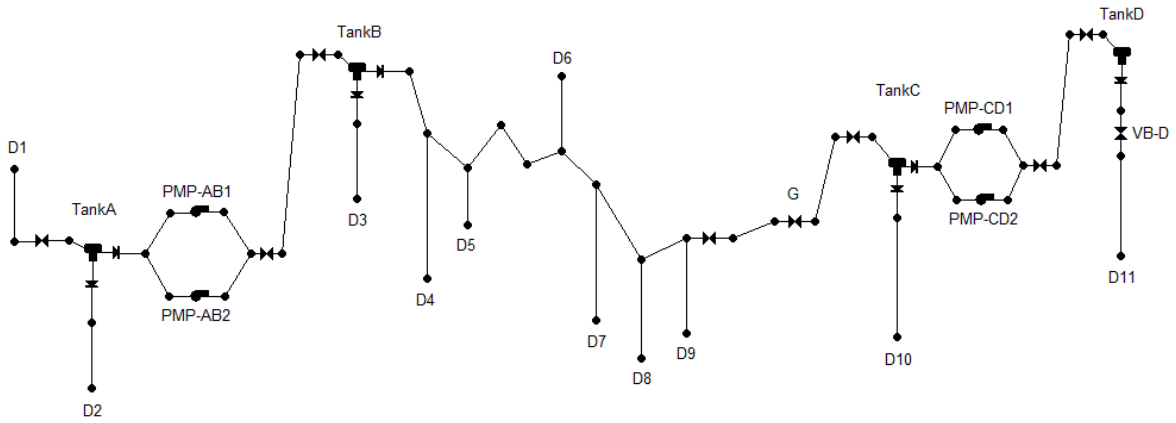


Figure 13.11: Simplified EPANET model of the Portuguese case-study used in this thesis.

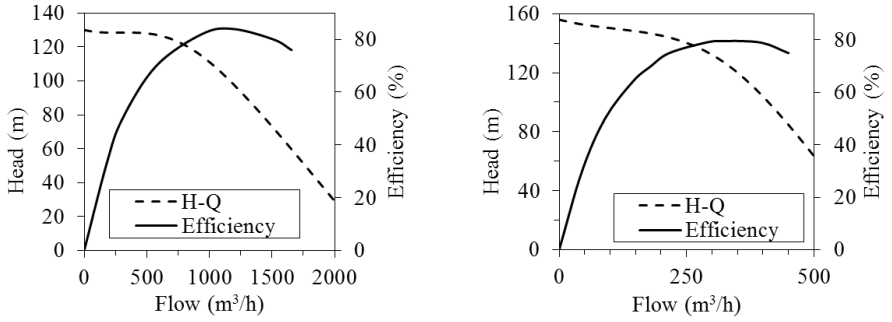
The network calibration was performed taking into account all data collected for a specific day of operation (a Saturday in a winter month) with the aim of simulating a real 24-hour period of operation. Through an iterative process, the simulation results were compared with the available measured data (mostly the water levels variation in tanks and the pumping stations energy consumption) and the model was adapted in order to achieve the best possible approximation. Since the calibration of hydraulic models is not part of the aims of this thesis, this topic is not further explored here[‖].

The efficiency and characteristic curves of the two different types of pumps represented in the model (PMP-AB and PMP-CD) are presented in Figure 13.12. The distinct demand patterns associated to each of the eleven demand nodes and the tariff considered for the simulated day are represented

[¶]Due to confidentiality reasons, the involved entities and specific details concerning the system analysed are not revealed.

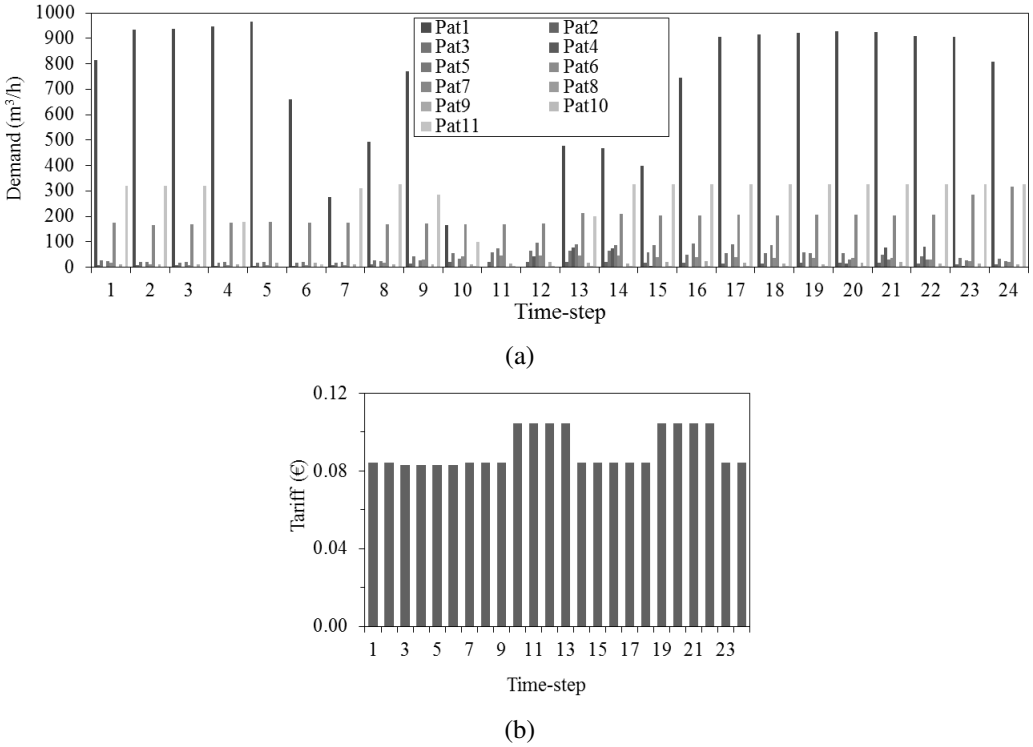
[‖]However, further information can be seen in Soares (2015).

in Figure 13.13. In comparison with the demand patterns related with the Richmond network, this network present significantly more variable demands. However, contrarily to the Richmond network, only one tariff is considered for all pumps.



(a) PMP-AB (b) PMP-CD

Figure 13.12: Efficiency and characteristic curves for the two types of pumps represented in the Portuguese network.



(a)

(b)

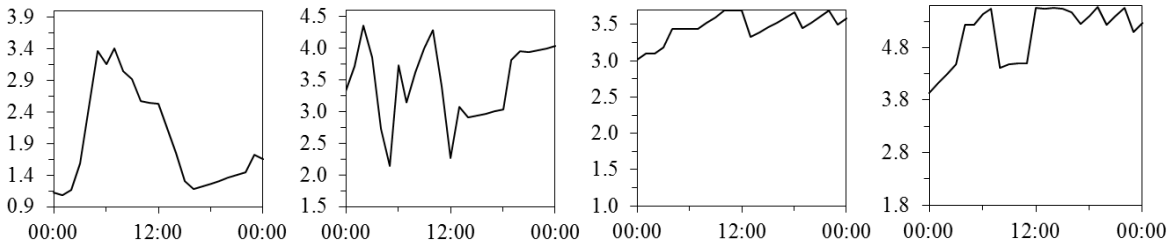
Figure 13.13: (a) Demand patterns and (b) tariff considered for the simulation of the Portuguese network.

The results of the network simulation for the selected day in terms of pumps usage and energy consumption are presented in Table 13.8. The variation of the water levels in each storage tank is represented in Figure 13.14.

From Table 13.8, it can be observed that only one pump per pumping station is operating. This fact is due to the simulated day being a winter day with low demand requirements. The network daily operating costs are 786.95 €, which corresponds to the value of the initial solution considered

Table 13.8: Results of the 24-hour simulation of the Portuguese network.

Pump	Utilisation (%)	Avg. Efficiency (%)	Energy (kWh/m^3)	Avg. Power (kW)	Peak power (kW)	Daily cost (€)
PMP-AB1	79.17	74.75	2.47	326.36	349.99	561.24
PMP-AB2	0.00	0.00	0.00	0.00	0.00	0.00
PMP-CD1	70.83	79.24	0.40	147.24	147.32	225.71
PMP-CD2	0.00	0.00	0.00	0.00	0.00	0.00
Total daily cost, $C_{\text{total}}(\mathbf{X}_0)$						786.95



(a) Tank A (b) Tank B (c) Tank C (d) Tank D

Figure 13.14: Variation of the water levels (m) in the tanks of the Portuguese network during the 24-hours of simulation considering the initial operational conditions. The minimum and maximum levels of each tank correspond to the minimum and maximum values presented in the vertical axis of each chart.

for optimisation. For more details on the initial conditions considered for this network, the EPANET input file is available in Appendix D.

13.2 Optimisation results

The presented optimisation results were obtained for simulation periods of one day divided into 24 time-steps. The starting points for the optimisation algorithms are the initial conditions previously described for each network. For the computation of the penalty functions, the following approach was followed for each distinct network: (i) in the case of the warning-related penalties, a value with one order of magnitude superior to the value of the daily operational cost was used as penalty coefficient; (ii) for the other penalties, a value with two orders of magnitude superior to the value of the daily cost was used for each penalty coefficient. Preliminary tests have shown effectiveness of the selected values.

From the five implemented optimisation algorithms, only results obtained with the NMSimplex, the DE, the PSO and the GA algorithms were presented. Since the tests performed with the ASA algorithm did not present any improvements, such results are not presented. The authors believe that such algorithm may be capable of provide satisfactory results since it has already demonstrated to performed well when applied to distinct optimisation problems (Ingber, 1996). However, due to the complexity of the algorithm, a more effective tuning should be performed in order to adjust the algorithm parameters to the type of optimisation problems that are addressed in this work**.

**ASA has over 100 options to provide robust tuning over many classes of nonlinear stochastic problems (Ingber,

13.2.1 Methodology validation and application of cascade optimisation techniques

In this section, both the van Zyl and the Richmond networks are used in order (i) to evaluate the performance of the proposed optimisation methodology and (ii) to test the sequential use of multiple algorithms (cascade optimisation techniques). The Particle Swarm Optimisation (PSO), the Differential Evolution (DE) and the Nelder-Mead Simplex (NMSimplex) are the algorithms used. The criteria used to change between algorithms in the cascade techniques is based on the pre-defined maximum number of iterations for each algorithm.

The Nelder-Mead Simplex, as a pattern-search algorithm, can be used as an attempt to improve the local search of the population-based algorithms (in this case, the sequence PSO+NMSimplex was tested). The sequential use of PSO and DE (both PSO+DE and DE+PSO) may be useful to overcome the limitations that each algorithm may present in comparison to the other. As already stated by Kachitvichyanukul (2012), these algorithms may be similar in terms of the exploration ability of the population, since the mechanism to generate new solutions are similar. However, the diversification in DE is higher than in PSO, since the best solution has no influence on the other solutions. At the same time, PSO has an higher tendency for premature convergence due to a fast clustering and consequent stagnation of the swarm. On the other side, in PSO, the best solution has more influence on population.

Results for the van Zyl network

In the case of the van Zyl network, besides van Zyl et al. (2004) that proposed and tested this network, other authors also tested distinct optimisation techniques for the control of this test network, as demonstrated in Table 13.9.

Using an optimisation method based on the control of the water levels in the tanks, van Zyl et al. (2004) applied an hybrid Genetic Algorithm (GA) combined with a hill-climber strategy which allowed to obtain a similar result compared with a simple GA but with a significant reduction in the number of function evaluations (*i.e.* 6 000 evaluations instead of the 100 000 required by GA). Considering random starting points, the optimal solutions obtained presented an average (from 7 runs) objective function value of 350.36 £/day with the pure GA and 348.58 £/day with the hybrid GA.

Later, López-Ibáñez et al. (2011) used an Evolutionary Algorithm (EA) to compare three different methods for the variables representation: (i) a traditional binary on/off (explicit formulation), (ii) a formulation based on the tanks levels control (implicit formulation) and (iii) another formulation based on pumps time controls (explicit formulation) distinguishing between absolute and relative time triggers. Results demonstrated that the explicit methods based on binary and relative time triggers representations were able to achieve the lowest values of the objective function. López-Ibáñez et al. (2011) also considered random initial solutions.

More recently, Hashemi et al. (2014) applied an Ant Colony Optimisation (ACO) algorithm to the same problem considering an explicit formulation for the pumps control optimisation and compared the use of fixed-speed pumps (only on/off variables) with the use of variable-speed (considering a speed range from 0 to 1). With a starting objective function value of 389 £/day (larger value than the ones obtained by van Zyl et al. (2004) and López-Ibáñez et al. (2011)), the methodology proposed by

2012).

Table 13.9: Comparison of the results obtained in the optimisation of the Van Zyl network, including the use of cascade techniques, with the results obtained by other authors (Coelho & Andrade-Campos, 2014b).

Authors	Optimisation technique	Method	Optimal cost (£/day)	Reduction	Evaluations [CPU time[a]]	Constraints (approach)
Van Zyl et al., 2004	GA	Tank level controls	350.36 (avg) 344.19 (best)	_	100 000	Tank levels and pump switches (cost penalties)
	Hybrid GA with Hill-climber (Hooke & Jeeves)		348.58 (avg) 344.43 (best)	_	6 000	
López-Ibáñez et al., 2011	EA	Binary	333.0 (avg) 324.7 (best)	_	6 000	Tank levels, pump switches and pressure (rank-based)
		Level controls	346.9 (avg) 337.2 (best)	_		
		Time controls (relative triggers)	334.1 (avg) 315.9 (best)	_		
Hashemi et al., 2014	ACO	Pump on/off	388.04	0.25 %	400 000	Tank levels and pump switches
		Pump speed	349.43	10.17 %	300 000	Tank levels
this work (Coelho & Andrade-Campos, 2014b)	**PSO** (n_P=200)	Pump speed and time controls	236.50 (avg) 230.64 (best)	**33.19 %**	200 000 [∼ 3 h]	Tank levels (cost penalties)
	DE (n_P=200)		289.23 (avg) 288.25 (best)	16.50 %	300 000 [∼ 17 h]	
	NMSimplex		342.84	0.70 %	9 412 [∼ 10 min]	
	PSO+NMSimplex		239.91 (avg) 237.63 (best)	31.17 %	126 919 [∼ 2.2 h]	
	PSO+DE		238.04 (avg) 235.73 (best)	31.72 %	275 000 [∼ 10 h]	
	DE+PSO		249.59 (avg) 239.73 (best)	30.56 %	275 000 [∼ 13 h]	

[a] Intel® Core™i7 processor, 3.40 GHz

Hashemi et al. (2014) was able to decrease the pumping energy costs to 388.04 £/day with fixed-speed pumps and to 349.43 £/day (a reduction of 10.17 %) with variable-speed pumps.

The methodology proposed in this thesis, based on the pumps operating times and speeds, was able to compute the most cost-effective solutions for the van Zyl network using the PSO and DE algorithms (Coelho & Andrade-Campos, 2014b). The results presented in Table 13.9 were obtained with an initial population of 200 individuals, from which one corresponds to the initial solution presented in Section 13.1.2 with a daily cost of £ 345.24 and the others were randomly chosen. The average cost reduction were superior to 33 % with the pure PSO and superior to 16 % with the pure DE. On the other side, the NMSimplex algorithm was only able to reduce the daily cost to £ 342.84 (0.7 %).

The algorithm with the higher performance (PSO) was tested sequentially with the NMSimplex (PSO+NMSimplex) in an attempt to improve the local search of PSO. However, no improvement was observed. From Figure 13.15, the performance of each single algorithm as well as each applied cascade technique can be analysed. A clear observation is that DE demonstrates a lower convergence compared to the PSO. While the PSO (both single and combined) was able to converge to an optimal

solution in less than 25000 function evaluations, the DE algorithm only starts to converge after 100000 function evaluations. The sequence DE+PSO was the only one improving the performance of the single algorithm (DE in this case).

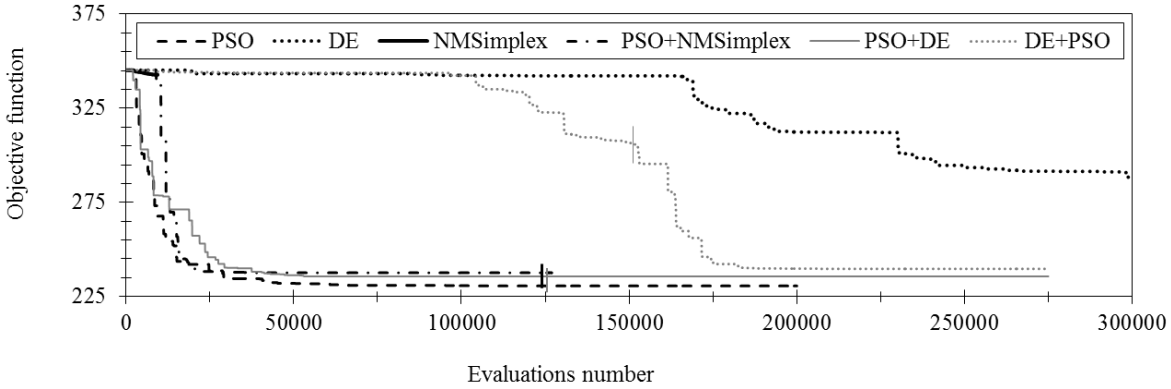


Figure 13.15: Performance of each individual optimisation algorithm compared to the cascade techniques applied to the operation of the van Zyl network. The moment of change between algorithms in the sequential optimisation is represented by a vertical bar that separates the corresponding line of evolution.

The best control solution for the daily operation of the van Zyl network obtained with the proposed methodology using the PSO (230.64 £/day) is represented in Figure 13.16, showing the controls (time and relative speed) obtained for each pump and the resulting water levels variation in the tanks.

Between the simulation hours 3 and 15, the pump 2B is operating at lower speeds and the pump 3B boosts its operation from 11 to 14 h. During the following 2 hours, only pump 3B is operating, which means that water is being pumped from tank A to tank B (as can be observed by the level increase in tank B). After that period, corresponding to the time of the day with lower energy prices, all pumps are operating at higher speeds until the tanks reach the required levels, as imposed by the constraints.

In terms of energy, considering the best operational solution, the results obtained for each pump are presented in Table 13.10. Comparing with the results of the initial solution presented (Table 13.3) the achievement of 33 % of cost reduction was obtained through the decrease in the percentage of utilisation of pump 3B (from 54.17 % to 50 %) and speed variations in the three pumps. On the one hand, the increase in the speed of pump 1A resulted in an increase of energy costs (from 20.95 to 29.69 £/day), on the other hand, the combination of higher and lower speeds in both pumps 2B and 3B resulted in decrease of energy costs (from 291.38 to 178.14 £/day and from 32.91 to 22.81 £/day, respectively).

The previous discussed solution was in fact the best solution obtained with the developed tool. However, it should be noticed that (i) van Zyl et al. (2004) and López-Ibáñez et al. (2011) restricted the number of function evaluations to 6000 while the best solution proposed was obtained for 200000 evaluations (or, at least, for 50000 evaluations, since the PSO algorithm stagnated at that stage, as shown in Figure 13.15), and (ii) Hashemi et al. (2014) limited the range of pumps speed to [0,1], meaning that no speeds above the nominal speed were allowed in their work. In the results obtained with the developed tool, the pumps speed range was set to [0,2]. In fact, considering a maximum relative speed of 2 may not be the most adequate value, since several variable-speed pumps are typically

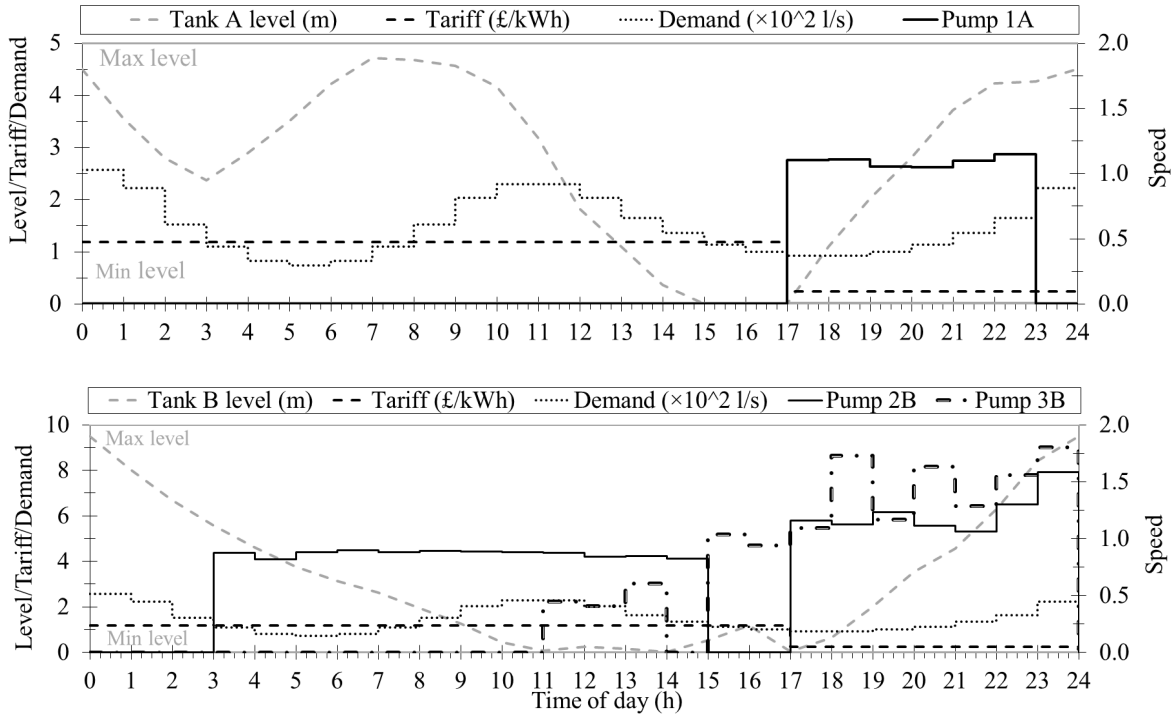


Figure 13.16: Operational conditions considering the best optimisation solution obtained for the van Zyl network.

Table 13.10: Energy results of the best solution obtained for the van Zyl network optimisation with PSO over 200 000 evaluations.

Pump	Utilisation (%)	Avg Eff. (%)	Energy (kWh/m^3)	Avg power (kW)	Max power (kW)	Daily cost (£)
1A	25.00	71.14	0.41	202.78	249.44	29.69
2B	79.17	73.42	0.32	190.39	740.56	178.14
3B	50.00	85.00	0.11	57.61	155.75	22.81
Total daily cost, $C_{\text{total}}(\mathbf{X}^*)$						230.64

allowed to operate until 150 % of nominal speed.

In order to obtain results capable of allowing a fair comparison with previous works, the van Zyl network was optimised considering distinct optimisation conditions. The number of function evaluations was restricted to 6 000 by reducing the swarm size (population size) to 50 and the maximum number of iterations to 120. The same calculations were performed for 3 distinct speed ranges, including (i) the range used in the previous results presented (0 to 2), (ii) the range used by Hashemi et al. (2014) (0 to 1) and (iii) a range more adequate to real pumps, but still, slightly close to the used by Hashemi et al. (2014) (0 to 1.2). The obtained results are presented in Table 13.11.

Table 13.11: Results for the optimisation of the van Zyl network considering only 6000 function evaluations and allowing different speed ranges for the pumps.

Algorithm	Evaluations	Speed range	OF (£/day)		CPU time[a] (min)	Reduction
PSO (n_P=50)	6 000	[0.0;2.0]	avg	264.49	12.09	23.39 %
			best	251.54	9.92	27.14 %
		[0.0;1.0]	avg	343.66	5.96	0.46 %
			best	342.56	5.85	0.78 %
		[0.0;1.2]	avg	277.71	8.70	19.56 %
			best	263.42	7.27	23.70 %

[a] Intel® Core™i7 processor, 3.40 GHz

An important conclusion obtained from the presented results is that the variation of the pumps speed for both lower and higher values can allow larger cost reductions than by uniquely reducing the speed. This happens because, when operating at higher velocities, even with some efficiency loss, the pumps are operating at larger values of flow discharge, which means that some water demands may be supplied in less time, taking advantage of the periods where the energy price is lower. The results obtained with the proposed methodology for the speed range [0;1] did not show improvements in comparison with the best results obtained by López-Ibáñez et al. (2011). However, by allowing the pumps to operate at speeds until 120 % of the nominal speed, such results are significantly improved, leading to energy cost reductions superior to 20 %.

In 6 000 function evaluations, the results obtained with the methodology proposed in this thesis were able to improve the ones obtained by Hashemi et al. (2014) (Table 13.9). Using the same number of evaluations (300000) could even improve such results. At the same time, the obtained results demonstrated that Hashemi et al. (2014) could possibly improve their results by considering a distinct speed range.

In average, the results obtained with the PSO algorithm in 6000 function evaluations considering the same speed range (0 to 2) presented a lower performance compared with the results obtained over 200 000 evaluations but, still, presented considerable improvements in comparison with the presented in previous works. The operational conditions as well as the energy results for the best solution obtained over 6 000 function evaluations are presented in Figure 13.17 and Table 13.12, respectively.

As shown by the percentage of pumps utilisation, in this operational solution (Table 13.17), the booster pump 3B is operating during a longer period when compared to the previous analysed solution (Table 13.10). However, the associated daily costs reveal to be inferior. This occurred due to the reduction in the energy consumed per cubic meter of pumped water (from 0.11 to 0.08 kWh/m^3).

Table 13.12: Energy results of the best solution obtained for the van Zyl network optimisation with the PSO algorithm after 6000 evaluations.

	Utilisation (%)	Avg Eff. (%)	Energy (kWh/m^3)	Avg power (kW)	Max power (kW)	Daily cost (£)
1A	25.00	66.35	0.45	298.14	478.72	43.65
2B	79.17	74.75	0.33	187.35	1005.06	186.54
3B	54.17	85.00	0.08	40.95	175.34	21.35
Total daily cost, $C_{\text{total}}(\mathbf{X}^*)$						251.53

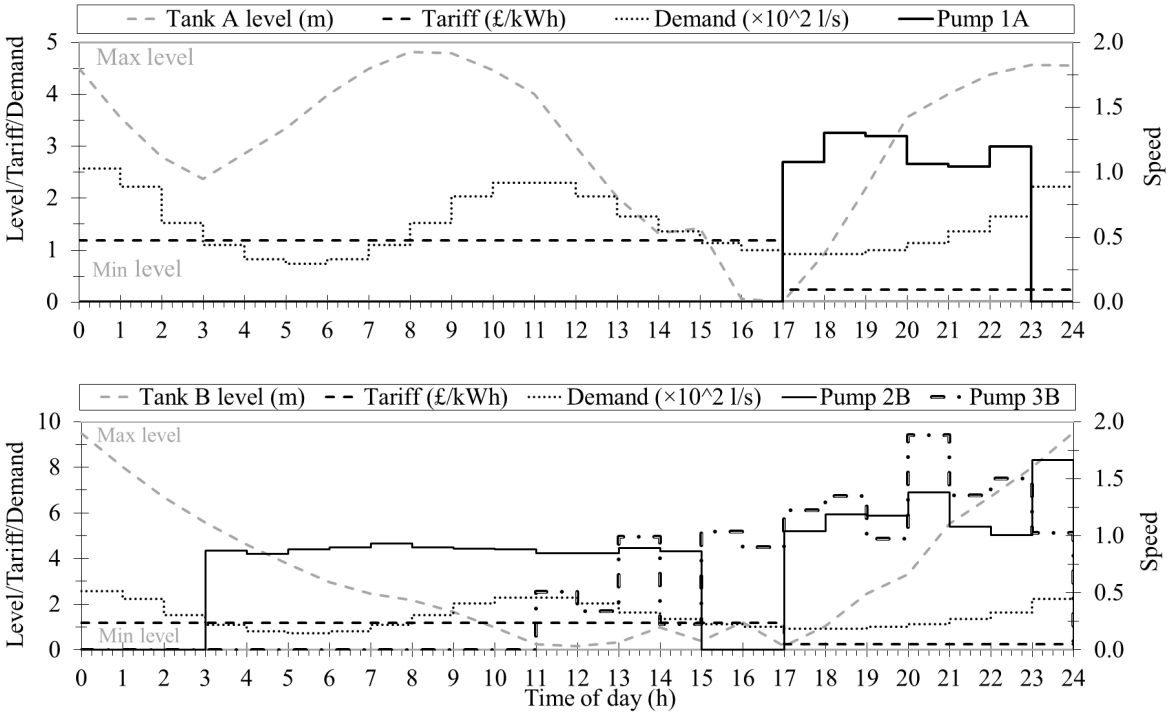


Figure 13.17: Operational conditions considering the best PSO solution obtained for the van Zyl network in 6 000 evaluations.

Pump 1A, besides presenting the same percentage of utilisation as the previous solution, has, in fact, higher associated operating costs. This is justified by the operation at higher speeds that, in this case, lead to a significant reduction in the average efficiency (from 71.14 to 66.35 %), explaining the higher energy costs (43.75 £/day compared to the previous solution of 29.69 £/day).

Figure 13.18 provides a representation of the operating points obtained for the pump 3B in the optimisation with the PSO over 6 000 evaluations. The corresponding characteristic curves modified by the speed are also represented. From such figure it becomes clear that, in some periods, the pump is in fact operating at both excessively low and excessively high values of speed, which, by the characteristic curves, do not seem realistic results. However, this is possible to be performed by the hydraulic simulator if the conditions are favourable in terms of head differences, as demonstrated in Figure 13.19, which represents the hour of the day where pump 3B is operating at 20 % of the nominal speed. At 2 p.m., the difference in elevations between the node N3 and the tank B is only 1.86 m. These results demonstrate the importance of the option included in the developed tool for changing the range of operation of the pumps. Both the minimum and the maximum speed allowed for the pumps (usually provided by the manufacturers) should be provided in this kind of optimisation problem in order to obtain valid results for the operation of real pumps.

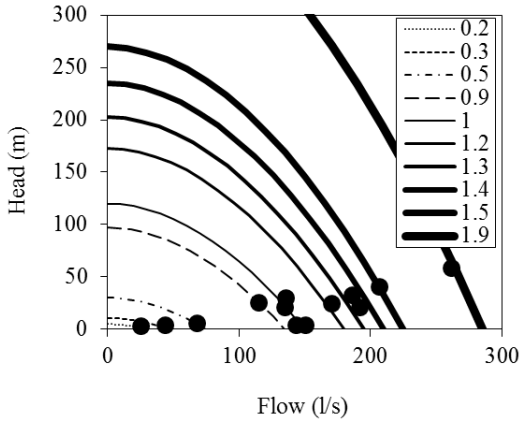


Figure 13.18: Pump 3B operating points from the best solution achieved by PSO over 6000 function evaluations in the optimisation of the van Zyl network.

None of the optimisation solutions presented for the van Zyl network provided evidences of advantages from considering operating time decision variables, t_{op}. In fact, both the presented solutions were possible to obtain uniquely by changing the relative speed of the pumps.

Results for the Richmond network

A similar procedure to compare the use of a single algorithm with the use of cascade techniques was also performed in the optimisation of the Richmond network (Coelho & Andrade-Campos, 2014b). The results are summarised in Table 13.13.

In the operational optimisation of the Richmond network, even with the high number of optimisation variables, both PSO and DE were able to achieve significant cost reductions. The lowest operational cost was obtained with the PSO (more than 21 % of reduction). However, the computational effort revealed to be superior when compared with the DE over the same number of function evaluations (300 000). On the other side, the DE only achieved 14.77 % of cost reduction.

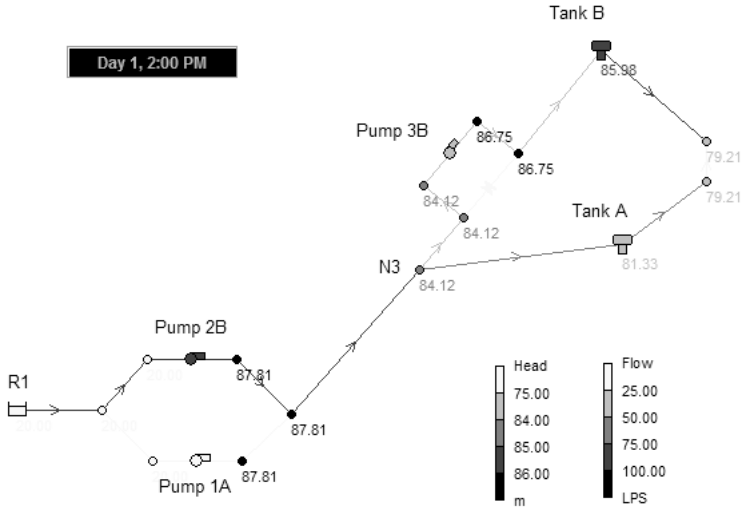


Figure 13.19: Operation of the van Zyl network at 2 p.m. demonstrating that pump 3B is operating at a low speed due to the low head difference that needs to overcome.

The sequential application of the PSO followed by the DE (PSO+DE) demonstrated to be advantageous uniquely in terms of computational time when compared with the PSO alone. However, the cascade technique DE+PSO reduced the costs in almost 22% in less time than the single PSO. As can be observed in Figure 13.20, the PSO was capable of achieving a lower value of the objective function by starting with the solution obtained by DE. PSO has in fact demonstrated to have a fast convergence but also to reach the stagnation very soon. For this reason, the sequential use of this algorithm multiple times (with less iterations) with DE (*e.g.* DE+PSO+DE+PSO) could provide some advantage.

Table 13.13: Optimisation results for the Richmond network comparing the use of single algorithms with the use of cascade techniques (Coelho & Andrade-Campos, 2014b).

Algorithm	Initial cost (£/day)	Optimal cost (£/day)	Reduction (%)	Evaluations number	CPU time[a]
PSO (n_P=50)	15633	12333	21.1	300 000	19 hours
DE (n_P=50)		13324	14.77	300 000	13 hours
NMSimplex		15620	0.08	3721	37 min
PSO + DE		12840	17.9	300 000	16 hours
DE + PSO		12212	21.9	300 000	16 hours

[a]Intel® Core™i7 processor, 3.40 GHz

Table 13.14 summarises the energy results obtained with the best solution achieved by PSO, the algorithm that presented the fastest convergence to the optimum according to Figure 13.20. Comparing with the initial solution (Table 13.5), it is noteworthy that no changes occurred in terms of pumps percentage of utilisation. In fact, for this network, the savings were obtained uniquely through an increase in the average efficiency of pumps operation due to the operation at distinct speeds (except for pumps 6D and 7F that slightly decreased their average efficiency).

Analysing Table 13.14 along with Figure 13.21, while for pump 1A there is no significant change

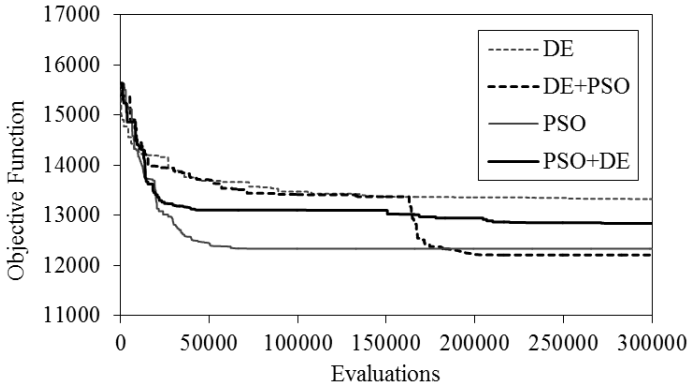
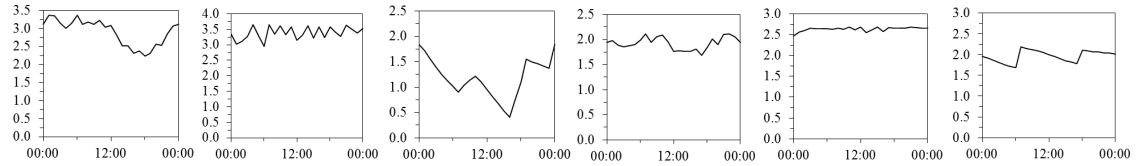


Figure 13.20: Evolution of the objective function for a representative solution obtained in the operational optimisation of the Richmond network considering cascade techniques and single algorithms.

Table 13.14: Energy results of the solution obtained for the Richmond network with the PSO algorithm.

Pump	Utilisation (%)	Avg. Efficiency (%)	Energy (kWh/m^3)	Avg. Power (kW)	Peak power (kW)	Daily cost (£)
1A	4.17	69.94	0.48	50.33	50.33	121.26
2A	100.00	71.46	0.41	51.10	86.11	6297.86
3A	100.00	48.42	0.13	15.49	34.87	1991.19
4B	50.00	54.74	1.86	18.79	58.54	1847.82
5C	29.17	63.98	0.40	4.87	6.98	336.31
6D	95.83	48.16	0.73	9.71	20.58	1715.47
7F	8.33	27.21	0.56	1.73	1.93	23.09
Total daily optimal cost, $C_{\text{total}}(\mathbf{X}^*)$						12333.00

in the daily cost, for pumps 2A and 3A the cost was reduced from 7603.65 to 6297.86 £/day and from 2536.51 to 1991.19 £/day, respectively. This occurred due to the large decrease in the energy consumption per unit of pumped water, revealing operations at lower values of flow, which are related with reduced speeds. Such type of operation led to a decrease in the water level in tank A (Figure 13.21) that was not observed in the initial solution (Figure 13.8). The relative speed values obtained for pump 2A varied between 0.8 and 1.1, while for pump 3A the range was [0.4;1.2]. The speed range obtained for the other pumps was [0.8;1.3].



(a) Tank A (b) Tank B (c) Tank C (d) Tank D (e) Tank E (f) Tank F

Figure 13.21: Variation of the water levels (m) in the tanks of the Richmond network during the 24-hours of simulation considering the optimal solution obtained with the PSO algorithm.

Although pump 4B also presented a reduction in the operating costs, the average energy consumption per unit of pumped water increased due to the operation at points with higher head values. At the same time, the control options obtained for this specific pump are not desirable due to the elevated number of pump switches observed (8 switches). For the other pumps, this number was never superior to 2 switches. From the water levels variation in tanks C and F, for instance, it is very simple to identify the 3 periods of operation of pump 5C and the 2 periods of operation of pump 7F, which correspond exactly to the periods of the tanks filling.

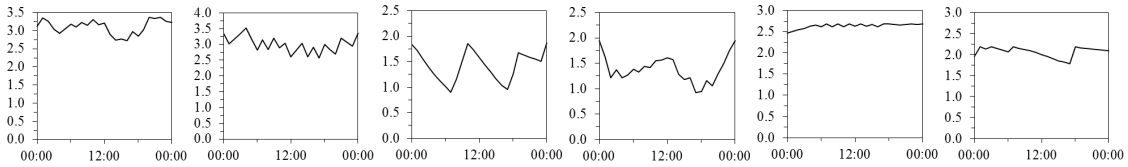
Similarly with the previously observed for the van Zyl network, this solution obtained with the PSO algorithm for the operation of the Richmond network do not demonstrate to take any advantage of the possibility of using time-steps inferior to 1 hour. The operation improvement was uniquely obtained by means of changing the relative speeds in certain periods of the day. However, analysing other solutions, it was verified that, for instance, the best solution obtained with the DE algorithm (also depicted in Table 13.13) took advantage of the variable time, t_{op}. Such solution reduced the daily energy costs by changing both the relative speed and the operating times of the pumps.

Table 13.15 and Figure 13.22 show the simulation results of the best solution obtained with the DE algorithm. In terms of the pumps percentage of utilisation, comparing with the solution obtained with the PSO (Table 13.14), a decrease in the time of operation is observed for all pumps except for pump 7F that increases the percentage of utilisation from 8.33 % to 13.14 %. Concerning the tanks water levels variation (Figure 13.22), the main differences are observed in tanks C, D and F. The control solution obtained for the pumps responsible for supplying such tanks (pump 5C, 6D and 7F, respectively) present changes both in terms of speed and time of operation. The three pumps present certain periods of operation inferior to the hourly time-step (in some cases the pumps operate less than half an hour) and the values of relative speed vary in the ranges [0.8;1.0] for the pump 5C, [0.9;1.3] for the pump 6D and [1.0;1.4] for pump 7F. The main drawback identified in this optimal solution obtained with the DE algorithm is related with the number of pump switches for some pumps, especially for pumps 3A (which also presented variable speeds and operating times) and 6D. The operational control of the remaining pumps was improved by taking advantage of only one type of

variable, (i) the relative speed, for pump 2A, and (ii) the operating time, for pumps 1A and 4B.

Table 13.15: Energy results of the solution obtained for the Richmond network with the DE algorithm.

Pump	Utilisation (%)	Avg. Efficiency (%)	Energy (kWh/m^3)	Avg. Power (kW)	Peak power (kW)	Daily cost (£)
1A	2.89	68.68	0.49	49.09	50.51	81.90
2A	100.00	68.57	0.89	55.39	61.10	7256.86
3A	79.79	52.53	0.49	17.92	33.98	1916.86
4B	47.33	63.77	0.15	18.29	20.33	1874.63
5C	22.86	70.29	0.39	6.14	6.71	332.13
6D	71.39	56.45	0.32	12.57	27.57	1822.43
7F	13.14	16.97	3.16	2.30	6.84	32.55
Total daily optimal cost, $C_{\text{total}}(\mathbf{X}^*)$						13317.35



(a) Tank A (b) Tank B (c) Tank C (d) Tank D (e) Tank E (f) Tank F

Figure 13.22: Variation of the water levels (m) in the tanks of the Richmond network during the 24-hours of simulation considering the optimal solution obtained with the DE algorithm.

13.2.2 Testing the approach for decision variables aggregation

The approach used in the developed numerical tool for the aggregation of the decision variables was tested in the van Zyl network in order to verify the influence in the feasibility of the obtained solutions and also in the computational time (Coelho & Andrade-Campos, 2014a). In the case of the Richmond network, no results of optimisation with aggregation are presented since the network does not allow any type of blocks aggregation.

The van Zyl network was optimised with PSO and DE algorithms with and without considering the aggregation of variables. Results for the best solutions of three runs are presented in Table 13.16. The evolution of the objective function average values are presented in Figure 13.23.

Table 13.16: Optimisation results obtained for the van Zyl network, including aggregation techniques.

Method	Optimisation algorithm	Optimal (£ /day)	Reduction (%)	OF evaluations	CPU time[a] (h)	Avg. Cost (£ /day)
Pump speed & time controls	PSO ($n_p = 100$)	227.77	35.7	100 000	2.64	234.38
	DE ($n_p = 100$)	292.75	17.3		2.38	299.53
Aggregated pump speed & time controls	PSO ($n_p = 100$)	232.14	34.4	100 000	6.23	233.70
	DE ($n_p = 100$)	304.78	13.9		4.45	312.31

[a]Intel® Core™i7 processor, 3.40 GHz

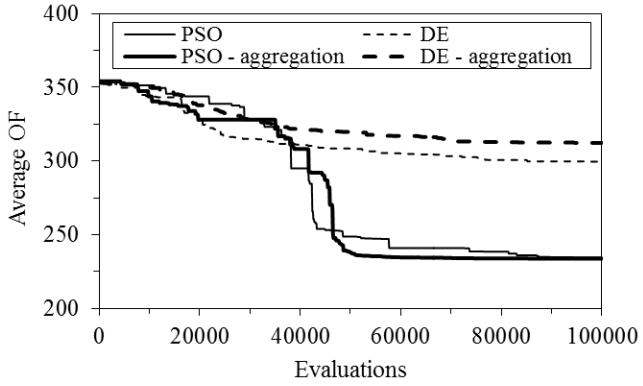


Figure 13.23: Evolution of the objective function considering the optimisation with DE and PSO with and without aggregation of the decision variables for the van Zyl network.

As can be observed in Figure 13.23, the starting point for both tested methods is not coincident with the initial solution previously stated in Section 13.1.2, with an initial daily cost of £ 345.24 . In fact, with the application of the variables aggregation approach, it is not possible to start from such previously defined initial solution. This is demonstrated in Figure 13.24, where, for the period of time between the 10 a.m. and the 12 a.m., the aggregation of two blocks of time is possible (same demand and same tariff in the two consecutive steps) but the control defined for pump 3B is different for each step. Since the variables aggregation is performed before the optimisation procedure, the pump control indicated in the first step is taken as the value for the aggregated block, which means that the resulting solution after disaggregation is the equivalent to the pump off from the 10 a.m. to the 12 a.m. instead of only from the 10 a.m. to the 11 a.m.. Such solution implies (a) a change in the pump operating time, and hence, in the energy costs and also (b) the violation of the tanks constraints, since the water level in tank B at the end of the simulation is inferior to the level in the beginning. For this reason, a different initial solution possible to be maintained even applying the aggregation approach was considered. The pump 3B was considered on between the 10 a.m. an the 12 a.m., leading to an initial daily cost of £ 354.09.

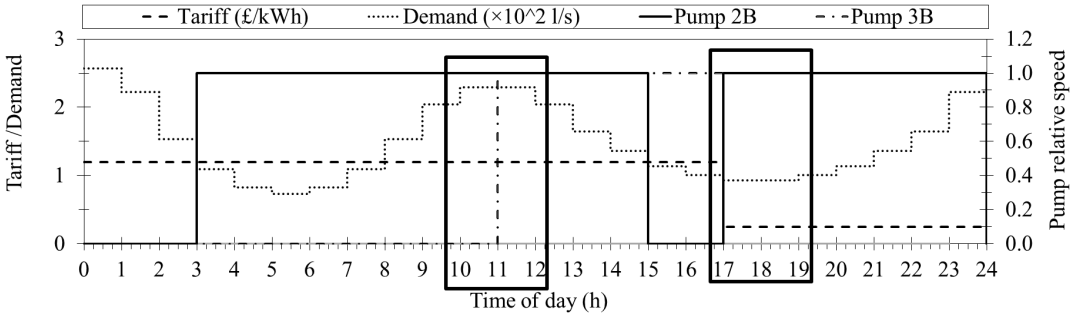


Figure 13.24: Demonstration of the periods of the day where the decision variables could be aggregated considering the initial solution presented in Section 13.1.2, which would result in two blocks of two aggregated steps each (the two consecutive periods with constant demand and tariff).

Comparing the results obtained with variables aggregation with the ones without aggregation, it can be concluded that the aggregation approach was not capable of improving the optimisation process. Furthermore, for this tested problem, the computational effort required with the use of such

proposed approach demonstrated to be superior to the required without aggregation, even with a reduction of the search space and the number of decision variables (less four variables). Analysing the report files generated by EPANET of the best solutions obtained with the PSO algorithm, it was verified that the optimal control solution found by the PSO using the approach for the variables aggregation required that EPANET performed 738 trials/iterations until converge to a stable hydraulic solution. In turn, the optimal control solution obtained without considering the aggregation approach required only 223 trials/iterations to converge, which corresponds to more than three times less computational effort. This means that the universe of solutions for the van Zyl problem with variables aggregation may be constituted of more complex (and hence, difficult to converge) than the universe of solutions for the problem without aggregation, which is hindering to take advantages from the aggregation module.

A brief discussion on the aggregation approach

As mentioned before, no possibilities of aggregation were possible in the Richmond problem. From the demand pattern considered for this network (Figure 13.7b), it is possible to observe that the values of demand differ in all steps. This means that the small differences between results are uniquely related with the probabilistic nature of the algorithms.

As it is currently implemented, the technique for variables aggregation may not be capable of detecting periods for aggregation since the demand values typically used in hydraulic models do not demonstrate fixed values over consecutive steps. This approach could possibly present some improvement by considering consecutive values of demand that do not differ from a small value close to zero, instead of values that are exactly the same (*i.e.* the difference between values are zero). In the case of the Richmond network, considering an absolute difference in demand up to 0.03, would allow 5 aggregation blocks, as demonstrated in Figure 13.25.

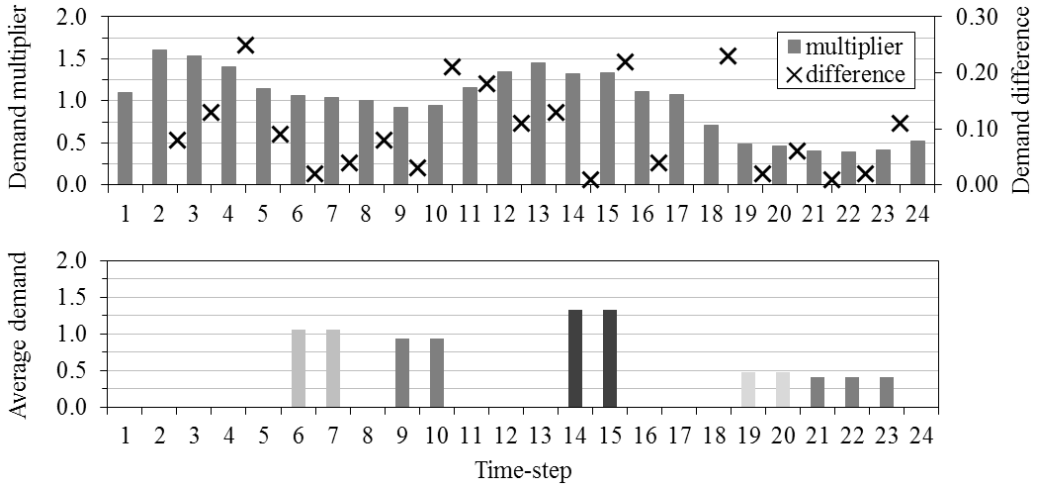


Figure 13.25: Demonstration of the possible blocks to aggregate the optimisation variables considering an absolute difference up to 0.03 between the demand values in consecutive steps.

13.2.3 Optimal operation of the Portuguese network

The daily operation of the Portuguese network was optimised with three different algorithms: PSO, DE and NMSimplex. The initial operational solution described in Section 13.1.5 was used as a starting point for the algorithms. The best and average results obtained in four runs of each algorithm are summarised in Table 13.17. The average performance of the algorithms is presented in Figure 13.26.

Table 13.17: Results of the optimisation of the Portuguese network with three different algorithms.

Initial cost	Optimisation algorithm	OF evaluations	Optimal cost	% cost reduction	CPU time[a] (min)	Average cost
786.95	PSO ($n_P = 50$)	50 000	729.23	7.33	161.50 [2.7 h]	736.88
	DE ($n_P = 50$)	50 000	709.47	9.85	284.87 [4.7 h]	725.60
	NMSimplex	2470	785.97	0.13	10.28	785.97

[a]Intel® Core™i7 processor, 3.40 GHz

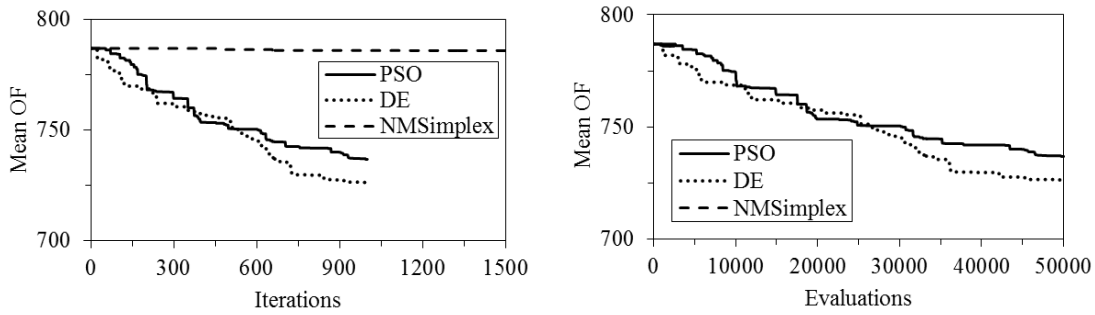


Figure 13.26: Evolution of the objective function (OF) as a function of (a) the number of iterations and (b) the number of evaluations in the optimisation of the Portuguese network with three different algorithms.

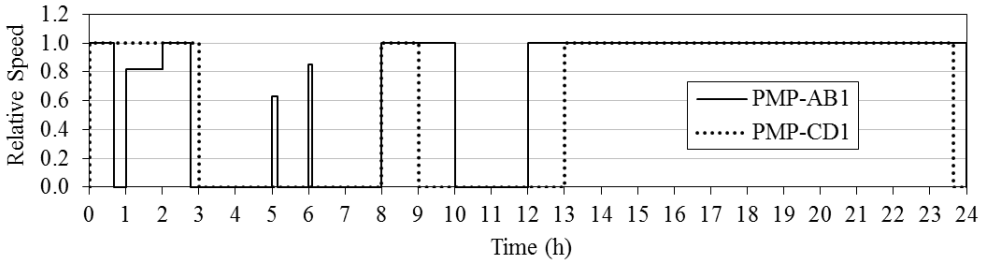
It should be noticed that the initial solution used for the Portuguese network corresponds to a typical day of operation that those in charge of the network's operations consider to be an already efficient solution since they have been improving the system control operation for several years. This particular network is an example of a network with variable-speed drives installed in the existent pumps that are only operating at fixed-values of speed (the points of maximum efficiency, also called the best efficiency points, BEP). According to that, the improvement of such operational conditions may reveal a challenge. Nevertheless, the developed numerical tool was able to provide significant reductions in the daily operating costs. Despite the NMSimplex algorithm have reduced the costs by 0.13 % in 10 minutes, which is not a notable result, the DE and the PSO algorithms were capable to obtain reductions of 9.85 % and 7.33 %, respectively, in 50000 function evaluations. These two algorithms demonstrated a similar performance. However, in average, the DE was able to converge to better solutions than PSO after 30000 evaluations of the objective function, as can be observed in Figure 13.26b.

The energy results, the optimal pump controls as well as the water levels variation in tanks ob-

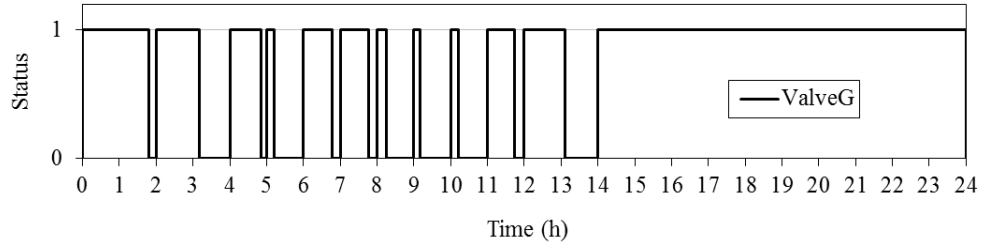
tained from the best solution found (with DE) are provided in Table 13.18, in Figure 13.27 and in Figure 13.28, respectively.

Table 13.18: Energy results for the best solution obtained with DE in the optimisation of the Portuguese network.

Pump	Utilisation (%)	Avg. Efficiency (%)	Energy (kWh/m^3)	Avg. Power (kW)	Peak power (kW)	Daily cost (€)
PMP-AB1	80.91	63.70	5.30	277.53	351.06	489.49
PMP-AB2	0.00	0.00	0.00	0.00	0.00	0.00
PMP-CD1	69.07	79.27	0.40	146.97	147.19	220.07
PMP-CD2	0.00	0.00	0.00	0.00	0.00	0.00
Total daily optimal cost, $C_{\text{total}}(\mathbf{X}^*)$						709.56



(a)



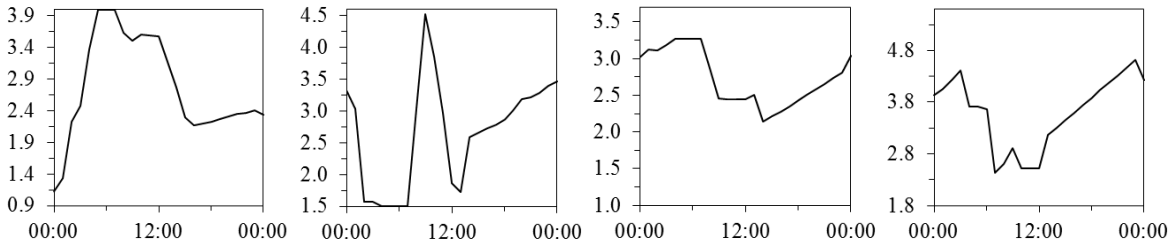
(b)

Figure 13.27: Optimal (a) pumps controls and (b) valve controls (1/0 correspond to open/closed status) obtained with DE in the optimisation of the Portuguese network.

The DE algorithm was capable of providing a solution in which the pumps that were initially off (PMP-AB2 and PMP-CD2) remained also off during the entire period of simulation.

Comparing the energy results of the optimised solution with the initial solution (Table 13.8), on the one hand, the percentage of utilisation of pump PMP-AB1 increased from 79.17 to 80.91 % and the efficiency decreased from 74.75 to 63.60 % due to the operation at lower relative speeds. However, even with the drop in efficiency, the operating costs were reduced from 561.24 to 489.49 €/day. On the other hand, the percentage of utilisation of pump PMP-CD1 decreased from 70.83 to 69.07 % and the average efficiency was kept nearly the same (79.27 % instead of the initial 79.24 %). Due to this decrease in the pump operating time, the average power consumption was reduced from 147.24 to 146.97 kW, leading to a daily cost reduction from 225.71 to 220.07 €.

The pumps operational solution that leads to the tank levels variation as represented in Figure 13.28 revealed to be a feasible solution since no constraints were violated. However, in terms of



(a) Tank A (b) Tank B (c) Tank C (d) Tank D

Figure 13.28: Variation of the water levels (m) in the tanks of the Portuguese network during the 24-hours of simulation considering the optimal solution obtained with DE algorithm.

number of switches, the obtained solution is not satisfactory for pump PMP-AB1 neither for valve G.

After analysing the best solutions obtained with the other two algorithms, it was verified that the PSO algorithm found a feasible solution both in terms of constraints violation and number of pump switches. The solution found by the PSO algorithm presents only three switches for pump PMP-AB1 and two switches for pump PMP-CD1, which were obtained uniquely by means of the variable relative speed, since the operating times were maintained in periods of one hour (the size of the time-step). The valve controls present four switches for the simulated day, which corresponds exactly to the number of switches of the initial solution.

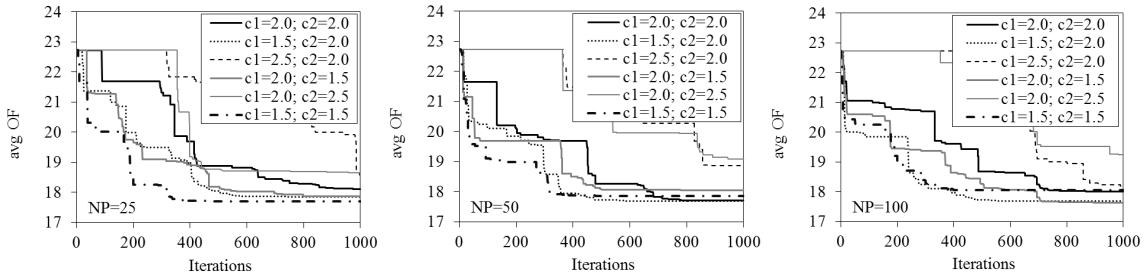
13.2.4 A sensitivity analysis study

Sensitivity analyses to the parameters of three of the algorithms implemented in the developed tool (namely, the Particle Swarm Optimisation - PSO, the Differential Evolution - DE and the Nelder-Mead Simplex - NMSimplex) were performed in order to measure the influence of using distinct parameters in the type of problems addressed in this thesis (water networks control optimisation). Such analyses were performed in the single-pump network (one variable-speed pump) and in the Van Zyl network considering two variable speed-pumps and one fixed-speed pump, which means that the algorithms are tested in optimisation problems with dimensions 48 and 120, respectively.

For the presented study, an approach similar with the proposed by Marchi, Dandy, Wilkins, and Rohrlach (2012) was followed. Thus, different population sizes were considered, namely, 25, 50 and 100 for the single-pump network and 50, 150 and 300 for the van Zyl network. The choice of such values were based in the work of Chen, Montgomery, and Bolufé-Röhler (2015) concerning the effect of dimensionality (number of decision variables) in PSO and DE algorithms. In PSO and DE algorithms, when dealing with low-dimensions problems (usually referred as $n < 50$), the recommended population size (n_P) is typically a value superior to the problem dimension ($n_P > n$). The experiments performed by Chen et al. (2015) demonstrated that $n_P < n$ is often required in high dimensions ($n > 50$). Since in evolutionary algorithms the use of a population size at least equal or superior to the problem dimension is common among researchers, in this study, three different values are tested: (i) a value near half the problem dimension, $n_P \approx n/2$, (ii) a value close to the problem dimension, $n_P \approx n$, and (iii) a value near twice the problem dimension, $n_P \approx 2n$.

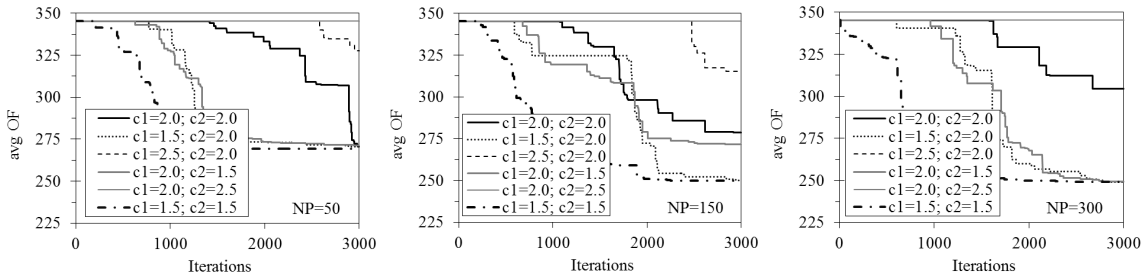
PSO parameters

The PSO parameters selected for perturbation were the cognitive and social acceleration parameters (c1 and c2, respectively), also called local and global learning factors. Fixing an initial combination for those parameters based on the commonly used values (c1=2.0 and c2=2.0), then the possible combinations considering an higher and a lower value (2.5 and 1.5, respectively) were tested for the three population sizes considered for each problem. Results in terms of the average algorithm performance for 2 runs are presented in Figure 13.29 for the single-pump network and in Figure 13.30 for the van Zyl network. The Results in terms of computation effort and cost reduction for the two tested networks are summarised in Table 13.19.



(a) n_P=25 (b) n_P=50 (c) n_P=100

Figure 13.29: Evolution of the mean objective function obtained from the optimisation of the single-pump network with the Particle Swarm Optimisation (PSO) algorithm considering three different population sizes (25, 50 and 100) and different combinations for the cognitive and social acceleration parameters, c1 and c2, respectively.



(a) n_P=50 (b) n_P=150 (c) n_P=300

Figure 13.30: Evolution of the mean objective function obtained from the optimisation of the Van Zyl network with the Particle Swarm Optimisation (PSO) algorithm considering three different population sizes (50, 150 and 300) and different combinations for the cognitive and social acceleration parameters, c1 and c2, respectively.

As expected, for both tested networks, the computational effort (CPU time) required by the PSO demonstrated to increase linearly with the increase of the population size. However, in terms of objective function improvements, there were no significant differences in the optimisation with different population sizes, neither in the single-pump network nor in the van Zyl network.

Globally, the parameters combination that revealed the best performance was c1=1.5 and c2=1.5, demonstrating a larger influence in the problem with high dimension (the van Zyl network), especially in terms of speed of convergence to the optimum. The obtained results may indicate that the previous

Table 13.19: Best results obtained with the PSO algorithm for each distinct population size considered in each tested network.

Tested network	Variables number	Population size	Iterations number	Initial cost	Best cost	% Cost reduction	Mean CPU time (min)
Single-pump (1VSP)	48	25	1000	22.73	17.68	22.21	10
		50			17.68	22.21	19
		100			17.58	22.68	41
Van Zyl (2VSP + 1FSP)	120	50	3000	345.24	250.01	27.58	295 [∼5h]
		150			249.23	27.81	869 [∼14h]
		300			246.90	28.48	1799 [∼29h]

optimisation results obtained for the van Zyl and the Richmond network, which were computed using the typical combination for the PSO parameters c1=2 and c2=2, could be improved by using other values for the parameters.

DE parameters

For the DE algorithm, the effect caused by the perturbation of the two following parameters was analysed: (i) the differential weight, F, and (ii) the crossover probability, CR. The reference set of parameters was selected due to the values proposed by Pedersen (2010) for various optimisation scenarios, *i.e.* F=0.6 and CR=0.95. Then, each one of the parameters was individually changed by a superior and an inferior value maintaining the other parameter. Figures 13.31 and 13.32 provide the evolution of the 2-runs average objective function for the single-pump network and the van Zyl network, respectively. An overview on the best results and the associated computational effort is provided in Table 13.20.

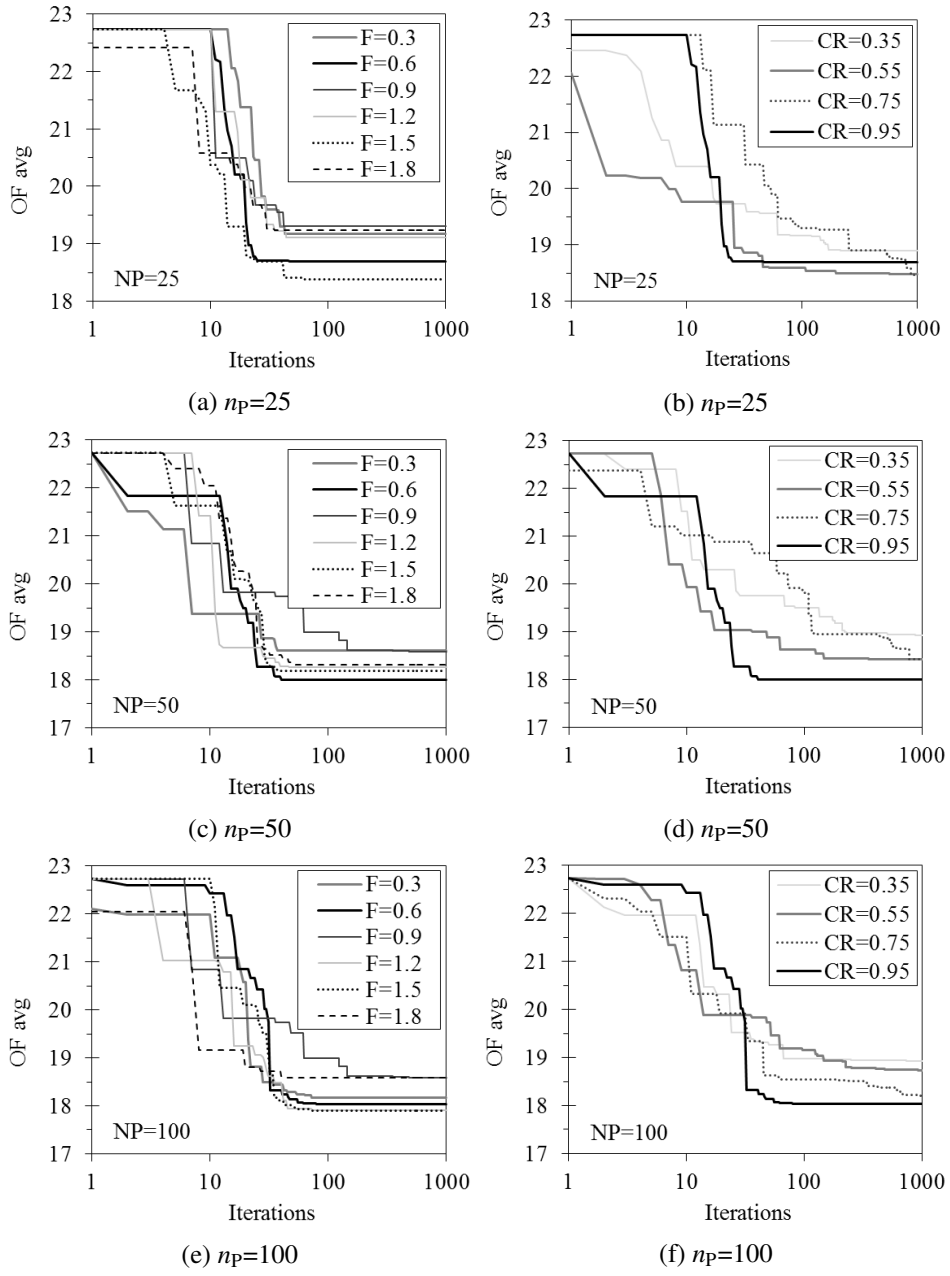
Table 13.20: Best results obtained with the DE algorithm for each distinct population size considered in each tested network.

Tested network	Variables number	Iterations number	Population size	Initial cost	Best cost	% Cost reduction	Mean CPU time[a] (min)
Single-pump (1VSP)	48	1000	25	22.73	18.09	20.43	14
			50		17.94	21.08	30
			100		17.85	21.49	66
Van Zyl (2VSP + 1FSP)	120	3000	50	345.24	294.55	14.68	198 [∼3h]
			150		271.57	21.34	541 [∼9h]
			300		251.98	27.01	1142 [∼19h]

[a]Using a processor Intel® Core™i5 CPU M460 @ 2.53GHz

Similarly with PSO, the DE algorithm also revealed a linear increase of the computational time with the increase of the population size. While in the case of the single-pump network the use of different population sizes did not cause influence in the optimum cost, in the van Zyl network, results demonstrated to be significantly improved with the increase of the population size.

Concerning the DE parameters, the value initially set for the differential weight (F=0.6) revealed to be, in fact, the most adequate for the single-pump. However, the performance of the DE in the van



(a) n_P=25 (b) n_P=25

(c) n_P=50 (d) n_P=50

(e) n_P=100 (f) n_P=100

Figure 13.31: Evolution of the mean objective function obtained from the optimisation of the single-pump network with the Differential Evolution (DE) algorithm considering three different population sizes (25, 50 and 100) and different values for the differential weight, F, and the crossover probability, CR.

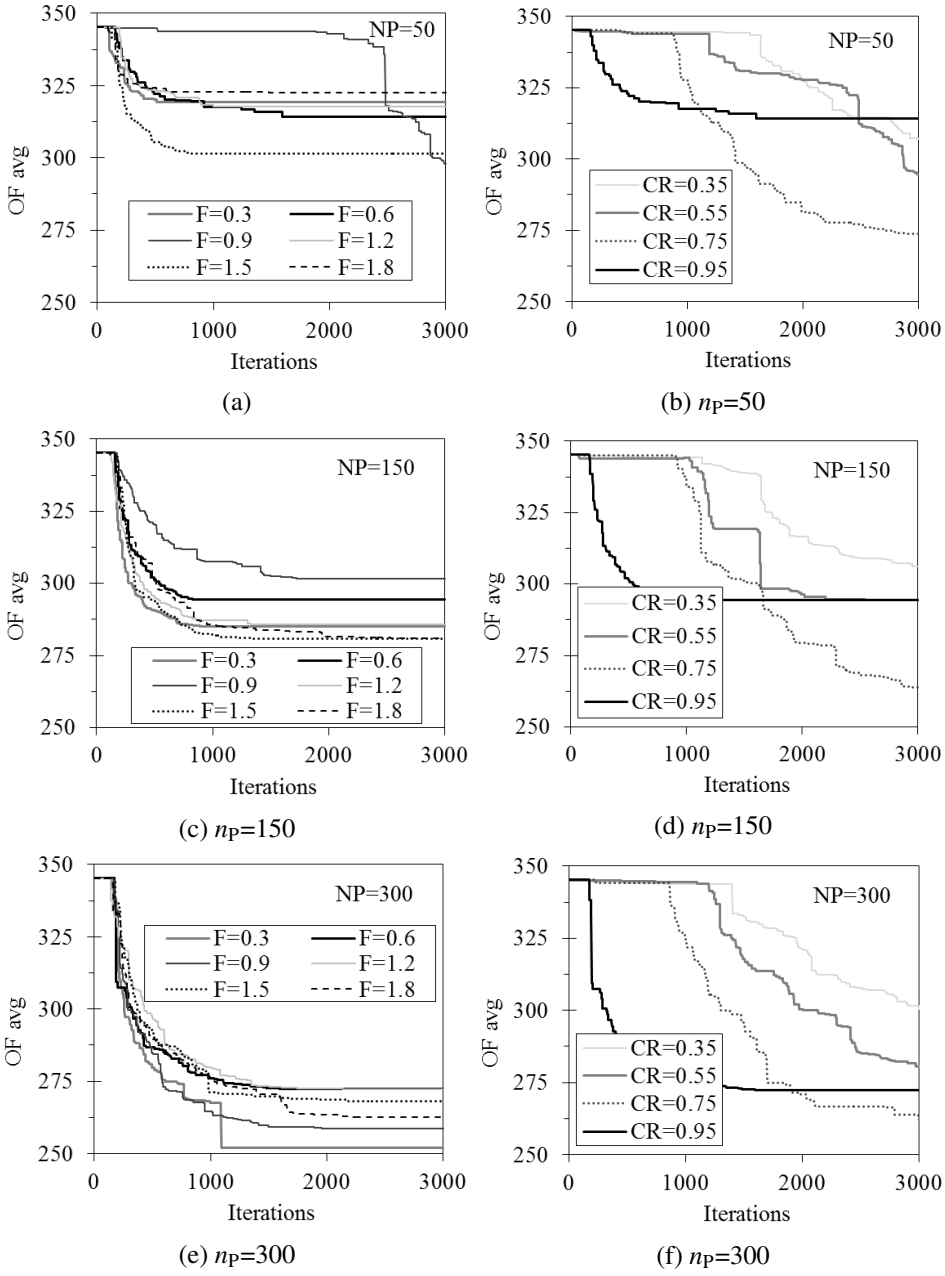
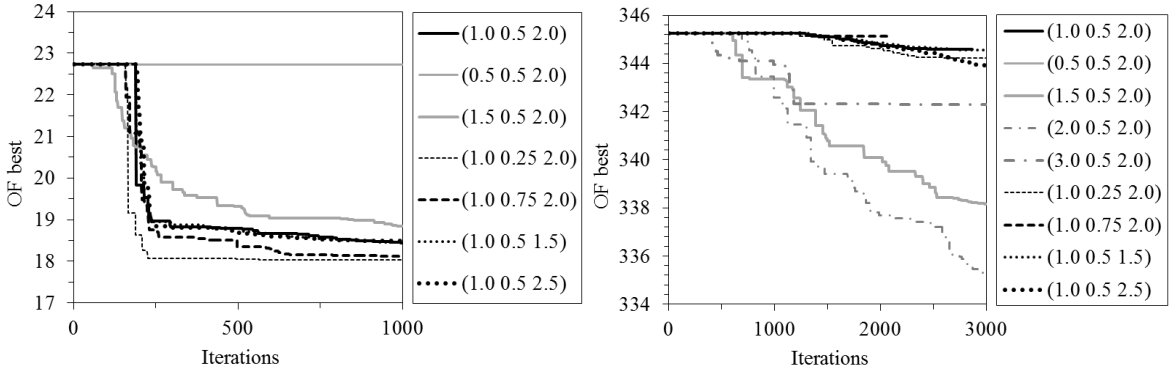


Figure 13.32: Evolution of the mean objective function obtained from the optimisation of the Van Zyl network with the Differential Evolution (DE) algorithm considering three different population sizes (50, 150 and 300) and different values for the differential weight, F, and the crossover probability, CR.

Zyl network was inferior when compared with the performance of the algorithm using other values of F. For a population size of 50 and 150, the best results were achieved with F=1.5, while for a population of 300, F=0.3 allowed to achieve the lowest value of the objective function. The value initially set for the crossover probability (CR=0.95) also demonstrates to be adequate when applied to the single-pump network. In the van Zyl network, such value of crossover probability provided the fastest convergence of DE to the optimum when compared with other parameters. However, for the three population sizes, the lowest average values of the objective function were obtained by using CR=0.75.

NMSimplex parameters

The effect of the parameters in the performance of the NMSimplex algorithm was tested by optimising the two networks considering distinct combinations of the following three parameters: (i) the reflection coefficient, α_r, (ii) the contraction coefficient, β_c, and (iii) the expansion coefficient, γ_e. Each parameter combination is represented by $(\alpha_r, \beta_c, \gamma_e)$. The set of parameters used as reference was based in the values typically used, *i.e.* (1.0, 0.5, 2.0). Each parameter was perturbed inferiorly and superiorly maintaining the other two parameters. Results of the parameters combinations are provided in Figure 13.33 and the best results achieved for each network are summarised in Table 13.21.



(a) Single-pump network (b) van Zyl network

Figure 13.33: Evolution of the objective function in the optimisation of two networks using the Nelder-Mead Simplex algorithm with distinct combinations for the reflection (α_r), contraction (β_c) and expansion (γ_e) coefficients represented, respectively, by (α_r β_c γ_e).

Table 13.21: Best results obtained by the optimisation of the single-pump and the van Zyl networks with the nelder-Mead Simplex algorithm.

Tested network	Variables number	Iterations number	Initial cost	Best cost	% cost reduction	Best coefficients (α_r β_c γ_e)	CPU time (min)
single-pump	48	1067	22.73	18.03	20.67	(1.0 0.25 2.0)	1.0
van Zyl	120	3001	345.24	335.14	2.93	(2.0 0.5 2.0)	6.8

The best results achieved by the NMSimplex algorithm in terms of cost reduction correspond to less than 3 % in the case of the van Zyl network but more than 20 % for the single-pump network.

NMSimplex is the algorithm that achieves the optimum in the most short period of time. However, this is also the algorithm that results in the lowest values of cost reduction.

The larger optimisation problem (the van Zyl network) demonstrated to be the most sensitive to the parameters perturbation. The two parameter combinations that provided the best results for the van Zyl were the (2.0, 0.5, 2.0) followed by the (1.5, 0.5, 2.0), i.e., the combinations that maintained the initial values of β_c and γ_e and increased the value of α_r up to 2.0. Note that $\alpha_r = 3.0$ was also tested but the algorithm demonstrated an inferior performance.

In the case of the single-pump network, all the tested parameter combinations, except for (0.5, 0.5, 2.0), allowed the DE algorithm to converge to satisfactory results. However, the higher cost reduction was obtained by considering the combination (1.0, 0.25, 2.0), which was followed by (1.0, 0.75, 2.0), both combination corresponding to perturbations in the contraction coefficient, from the initial value $\beta_c = 0.5$.

13.2.5 Optimising a network under fire flow conditions

The Walski network, described in Section 13.1.4, was optimised with four of the implemented algorithms, namely the PSO, the DE, the NMSimplex, and the GA. The first three algorithms were tested considering the most adequate parameters according to the sensitivity analysis study previously presented. In the case of the PSO algorithm, c1=c2=1.5 and n_P=50; for the DE, F=1.5, CR=0.95 and the same population size (for a fair comparison of results); and, finally, for the NMSimplex the parameter combination (2.0, 0.5, 2.0) was used. The obtained results and the performance of the algorithms are presented in Table 13.22 and in Figure 13.34, respectively. For the GA, the values of crossover and mutation probability used were 0.9 and 0.01, respectively. The results for PSO, DE and GA were obtained over a total of 4 runs performed for each algorithm.

Table 13.22: Optimisation results for the Walski network with four different algorithms: Particle Swarm Optimisation (PSO), Differential Evolution (DE), Nelder-Mead Simplex and Genetic Algorithms (GA).

Initial cost	Algorithm	OF evaluations	Best optimum cost	% cost reduction	Mean optimum cost	Mean CPU time (min)
324.36	PSO (n_P=50)	50 000	204.77	36.87	211.20	48.43
	DE (n_P=50)	50 000	212.99	34.34	242.48	62.22
	NMSimplex	11 296	287.90	11.24	287.90	10.57
	GA (n_P=50)	50 000	200.49	38.19	273.49	60.39

The solution with the lowest associated cost of operation was obtained with the GA (a reduction of more than 38 % comparing with the initial solution). In average, this algorithm took 60 minutes to converge to the optimum. In turn, the NMSimplex algorithm, besides having presented an inferior performance in terms of the value of the objective function, was able to converge to an optimum that resulted in more than 11 % of cost reduction in just 10 minutes. At the same time, considering only the first 10000 evaluations of the objective function, the NMSimplex algorithm demonstrated a faster convergence and better performance than both the GA and the PSO algorithms. The DE was the algorithm that performed better for the first 10000 evaluations.

Considering the average results (for the 4 runs), the PSO demonstrated the best performance by

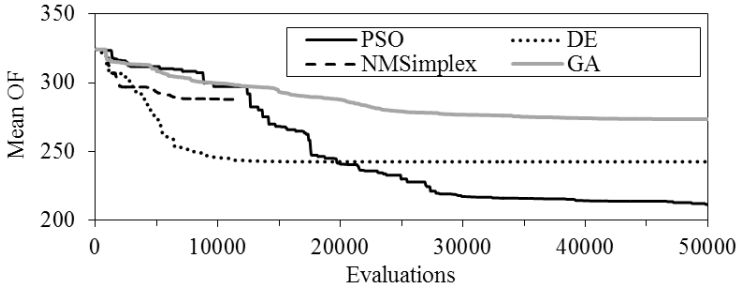


Figure 13.34: Evolution of the mean value of the objective function obtained in the optimisation of the Walski network with four different algorithms.

reaching optimum solutions with an average associated cost of 211.2 €/day. At the same time, the PSO converges to the optimum in a lower CPU time in comparison with both GA and DE. In average terms, the performance of GA decreases considerably.

The energy results for the simulation of the best operational solution obtained with the PSO (the algorithm that presented the best results in average) are presented in Table 13.23. The simulated operational conditions, including the pumps controls and the variation of the water level in the tank, are presented in Figure 13.35.

Table 13.23: Energy results for the best solution obtained in the optimisation of the Walski network with the PSO.

Pump	Utilisation (%)	Avg. Efficiency (%)	Energy (kWh/m^3)	Avg. Power (kW)	Peak power (kW)	Daily cost (€)
PMP-1	58.33	75.18	0.57	210.31	582.49	146.46
PMP-2	25.00	75.98	0.24	69.76	164.23	58.3
Total daily optimal cost, $C_{\text{total}}(\mathbf{X}^*)$						204.77

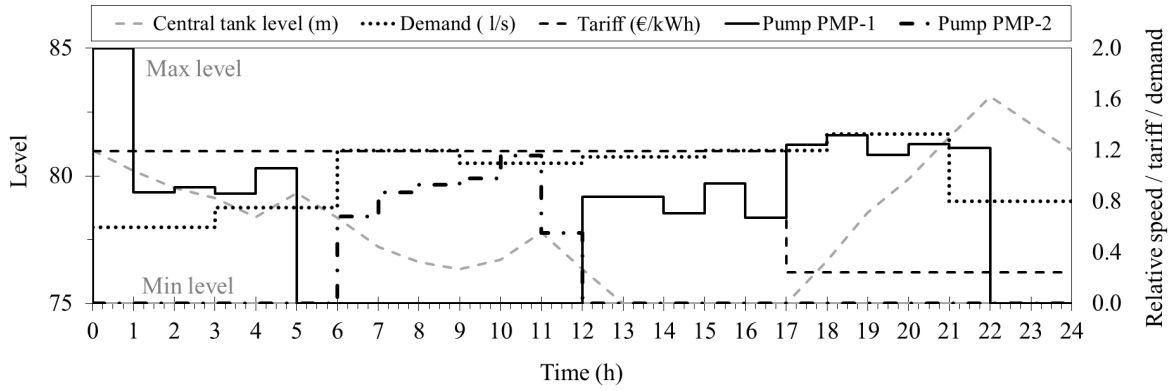


Figure 13.35: Operational conditions of the Walski network considering the best solution obtained with the PSO algorithm.

Even with a reduction in the average efficiency and an increase in the energy consumption per unit of pumped water, the daily operational costs were reduced from 237.22 to 146.46 € by reducing the pump speed in periods of time where the tariff is superior and increasing the speed in the cheaper periods.

For pump PMP-2, both the energy consumed per unit of pumped water and the average and maximum power were reduced (also due to the speed variation), leading to a reduction in the daily costs from 87.14 to 58.3 €.

From Figure 13.35, it is possible to see that, by the time of the fire occurrence (from 11 to 14h), the water level in tank drops until the minimum limit allowed (75 m) and remains near the minimum level until the period with lower energy cost, where the speed of the pump PMP-1 is increased and the pump continues in operation until the tank reaches the required level to finish the day with the same water level as in the beginning.

Similarly with the previous optimisation problems, the solutions obtained with the PSO algorithm demonstrate a tendency to result uniquely in variations of the speed variables. Analysing the solutions obtained with the other algorithms tested in this problem, both the GA and the DE algorithms demonstrated variations in terms of speed and time of operation of both pumps, as well as variations in the valves opening times. As previously pointed, the main drawback of such obtained solutions is still related with the high number of pumps switches. The solution obtained with the NMSimplex algorithm demonstrate to be similar with the solutions obtained with the PSO algorithm by changing only the relative speeds of the pumps.

14. Predicting water demands in a Portuguese case-study

Four water demand data sets are used to develop short-term forecasting models including naïve models, exponential smoothing and artificial neural networks (ANN). The influence of accounting anthropic and meteorological variables to ANN-based models is also analysed.

14.1 Case-study description

In this work, the idea of forecasting water demands emerged from the possibility of preparing and adjusting the operation of a WSS in a more efficient way, avoiding unnecessary operating costs.

Figure 14.1 provides a specific scheme of the Portuguese network, previously presented in Chapter 13, for a demonstration of the data measurement zones.

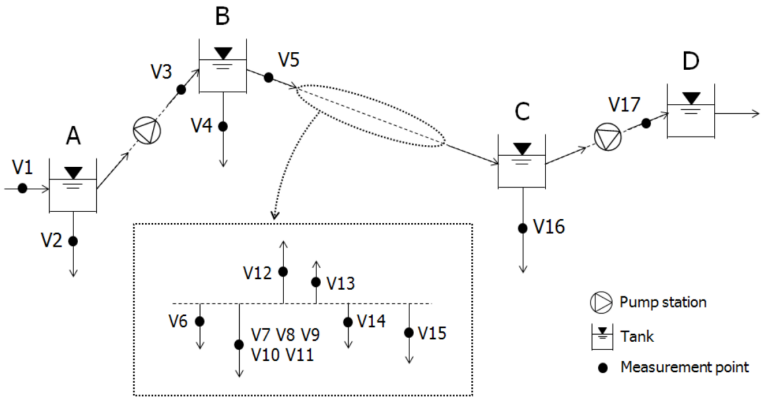


Figure 14.1: Simplified representation of the Portuguese water network of this study.

Data, from August 2012 to July 2013, measured in delivery points from which the consumers are supplied by gravity (V6 to V15) as well as tanks inlet (V1, V3 and V17) and outlet (V2, V4, V5 and V16, where only V5 is not a delivery point) were provided by the water utility responsible for the management and operation of the system*. Data from tank D is not available since it is managed by

*Due to confidentiality reasons, the water utility and the details concerning the system analysed are not revealed.

a distinct water utility.

The collected data were provided in the format of accumulated volumes of water, in cubic metres (m^3), measured in time intervals of 10 minutes.

The type of water consumers of this case-study represents a mix of domestic, agriculture and industrial consumers.

Besides the historical data of delivered water, some meteorological data, such as temperature, relative humidity and rainfall occurrence, was obtained from the nearest meteorological station in the area[†]. All this data was collected in hourly time intervals, during the same period (Aug 2012 to Jul 2013).

No information from experts, such as explanations for failures or unexpected occurrences, is available. This means that all the analysis of water demands is based on interpretations on the available historical data and meteorological effects.

The objective of this work is to evaluate the performance of distinct forecasting models for distinct delivery points of the described case-study considering the influence of distinct sets of input data (including historical demands, anthropic and meteorological variables). The forecasting models time-scale is hourly (same time period commonly used when dealing with operational control problems) and it is intended to evaluate the performance of each model when forecasting the next 24 hours of water demand.

14.2 Data selection

For this case study (Figure 14.1), there are enough available data to develop a model in order to predict future supply needs for both tank A and tank B (historical data measured in points V2, V4 and V5). Although it could be preferable to use data from each delivery point from V6 to V15 and V16, data from tank D outlet is still missing, which means that part of the demand needs for this network would be discarded. At the same time, from the time plots of each data point, it was observed that data collected at point V10 presented several inconsistencies and a lot of missing data that were not possible to interpret (possibly caused by failures in measurement equipment or communication system). For these reasons, it was decided to analyse in this work the data from V2, V4, V5 and V16. It is expected more difficulties in demand forecasting for point V5 since it does not represent a delivery point but, instead, the sum of the delivery points V6 to V15, V16 and the outlet water of tank D. Moreover, such data may be influenced by the operation of the pumping station that pumps water from tank C to tank D, which may be hiding possible demand patterns.

14.3 Data analysis and pre-processing

After plotting the time series of the raw data as provided by the water utility (see examples of some data sets in Figure 14.2) it became clear that in most of the cases, several failures occurred over the year, *i.e.* the observations of accumulated water volume where not always increasing over time as expected.

[†]Data provider: `http://freemeteo.co.uk/`; data source: `http://www.noaa.gov/`.

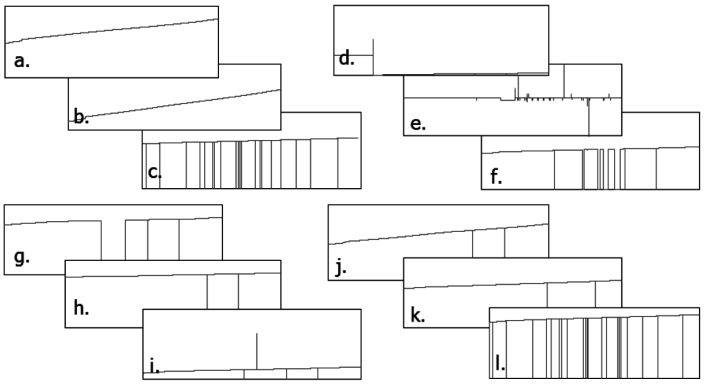


Figure 14.2: Examples of the time series plots of the raw data of the accumulated water volumes (in m^3) measured in 10-min time intervals at distinct points from August 2012 to July 2013.

Unlike the first two time series plots, a and b, that apparently presented data without outliers, all other examples needed to be cleaned. In plots c, f to h and j to l, several occurrences with the observations set to zero are identified as extreme outliers. This may probably be related to interruptions in data collection.

In order to clean the data, the first step was the detection and removal of outliers. The method used was based in the interquartile range of each data set (see, for example, outliers detection in Natrella (2010)), rejecting values inferior to the lower quartile (lower outlier boundary) and superior to the upper quartile (upper outlier boundary).

Once the outliers were removed, it was possible to identify other type of data failures resulting from the counting re-initialisation of the measurement device. Figure 14.3 provides a representation of this type of occurrence. All data sets presenting this type of occurrence were corrected by adding the value of the last measure (device limit) to the initialised values.

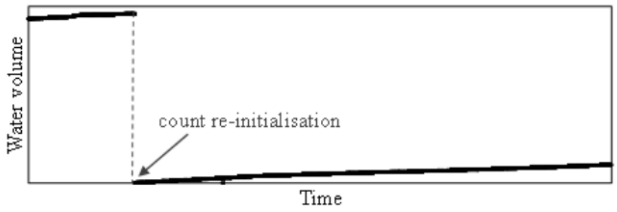


Figure 14.3: Representation of a possible occurrence in collected data. After reaching the limit of the measurement device, the counting starts again from zero.

An analysis to the amount of missing data was also performed. For all data sets, it was verified that in the first two months of data (August and September of 2012), more than 40 % of data were missing. For this reason, it was only considered the data from 21/09/2012 to 31/07/2013. All the other missing data identified represented only 0.4 % for V2 data set, 0.5 % for V4 and V16 data sets and 0.8 % for V5.

After correcting the mentioned particularities observed in data, since the measurements were performed in 10-minutes intervals, the corresponding values for fixed steps in hourly intervals were computed using data linear interpolation.

Finally, in order to obtain the hourly water demands (WD, in m^3/h), the differences between each

measured hour were computed. This allowed transforming the initial time series (V2, V4, V5 and V16) that presented a linear trend (water volume increasing linearly with time) into stationary series (WD2, WD4, WD5 and WD16).

A statistical overview of the water demand time series, including mean, standard deviation, maximum and minimum values, is presented in table 14.1.

Table 14.1: Statistical information concerning each water demand time series data set (from 21/09/2012 to 31/07/2013).

Data set	WD2	WD4	WD5	WD16
Total Observations	7536	7536	7536	7536
Mean, $\bar{y}$ (m^3/h)	11.62	37.73	441.31	2.67
Standard deviation, σ (m^3/h)	5.30	16.80	157.24	1.82
Maximum, y_{max} (m^3/h)	34.67	97.71	949.32	14.42
Minimum, y_{min} (m^3/h)	0.00	0.00	0.00	0.00

With the water demand series ready to be used for the development of forecasting models, it is important to analyse other available data in order to decide the predictor variables that may be included in the models.

After analysing the patterns revealed by the time series plots, it was verified that for different months, different patterns were presented, as well as in different days of the week. Thus, an analysis to the influence of anthropic variables (month, day of the week and hour of the day) was performed.

For the variable *Hour* (H), numbers from 0 to 23 were used to represent the 24 hours of a day. Concerning the variable *Day of the week* (D), numbers from 1 to 7 were used to represent Monday to Sunday and the same was done for the variable *Month* (M) using the numbers 1 to 12 (January to December).

In a first step, the Pearson correlation coefficients between the water demand data sets and the anthropic variables under consideration were computed. Results of such coefficients are provided in Table 14.2.

Table 14.2: Pearson correlation coefficients measured between the distinct water demand sets (WD) and the considered anthropic variables *Day of the week* (D), *Month* (M) and *Hour of the day* (H).

	D	M	H	WD2	WD4	WD5	WD16
D	1.000						
M	0.005	1.000					
H	0.000	0.000	1.000				
WD2	0.000	0.120	**0.605**	1.000			
WD4	0.047	0.116	**0.650**	**0.915**	1.000		
WD5	0.094	-0.057	0.130	0.107	0.120	1.000	
WD16	0.075	-0.008	**0.650**	**0.836**	**0.894**	0.069	1.000

Observing the gray area represented in Table 14.2, it is clear that the variable that presents higher correlation with the water demand series is the variable *Hour*, especially with WD2, WD4 and WD16. However, it was also expected some relationship with the other variables, which was not observed with these results.

Although the Pearson correlation (a quantitative sensitivity parameter) is often used by researchers for the choice of the variables to include in their forecasting models, such measure provides only information about the linear relationship between variables (Hamby, 1994). This means that, other type of relationship may be undetected with this approach. For this reason, it was decided to analyse the scatter plots (a qualitative sensitivity parameter) for all variables in order to reveal other possible relationships. Such plots are represented in Figure 14.4.

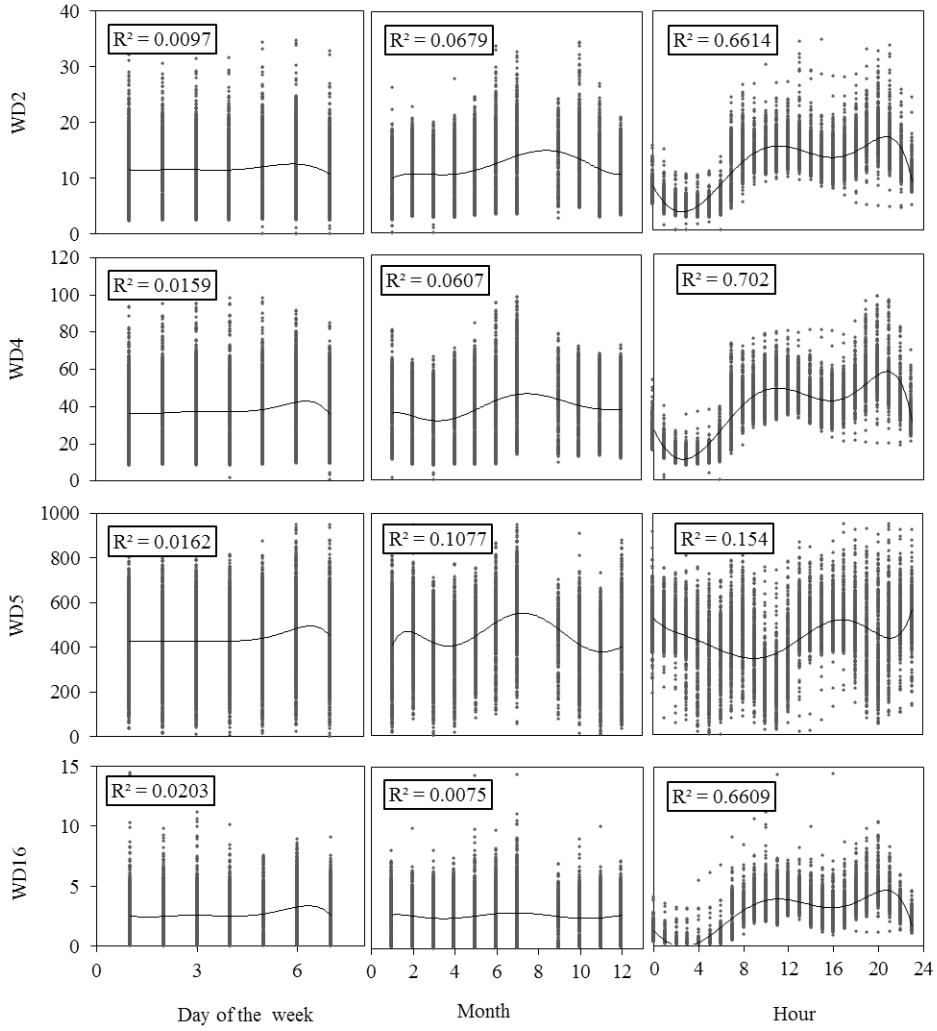


Figure 14.4: Scatter plots showing the relationship between the water demand time series (in m^3/h) and the anthropic variables *Hour* (H), *Day of the week* (D) and *Month* (M). Adjusted 6^{th}-order polynomial trend lines and the squared correlation coefficients are also represented.

In fact, according to Figure 14.4, the relationships between the water demands and the anthropic variables are not linear. The variable *Day of week* presents the weakest relationships with water demand. However, analysing, for instance, the relationship of this variable with WD5, it is possible to observe that higher water demands occur during the weekends.

Analysing the variable *Month*, it is also notorious the higher water demands for specific months (such as expected in the summer months, from June to September[‡]).

[‡]It should be noticed that data from August to mid-September is missing in the data sets, which may hide the

By adjusting, for example, a polynomial trend line instead of a linear one, the correlation coefficients between the variable *Hour* and the water demand significantly increases. All the correlation coefficients obtained from these scatter plots are presented in figure 14.5 for a faster analysis. The anthropic variables with higher correlations are marked in the figure with dashed lines and were the ones selected to be tested in the forecasting models of this work.

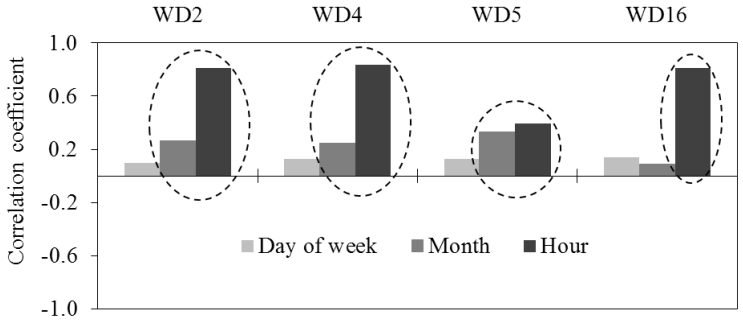


Figure 14.5: Correlation coefficients (from a polynomial trend) between the water demand in each data set and the anthropic variables. The variables signed with the dashed lines were the ones selected for the forecasting models development in this work.

From the results presented in Table 14.2, it is also worth to mention the strong relation between the water demand series and the water demand in the neighbouring points, *i.e.* WD2 has a strong correlation with both WD4 and WD16 and even WD4 present a high correlation with WD16. The scatter plot matrix provided in Figure 14.6 clearly shows these linear relationships. Such observations might mean that the inclusion of these variables (past water demands observed in neighbouring areas) in the forecasting models can be beneficial. Although not found in the literature the use of such variables for water demand forecasting, in this work, the forecasting models will also be tested including them.

Concerning the weather variables (temperature, relative humidity and rainfall occurrence), it is also important to perform an analysis to the time series to deal with possible outliers and missing data.

Although no outliers were identified, a large amount of the available data was missing. In the period considered for the water demand data (21/09/2012 to 31/07/2013), around 30 % of data were missing. After plotting the time series, it was observed that almost all data for the months of October and November were missing. Considering only the data from 3/12/2012 to 31/07/2013 (last 5761 observations) the amount of missing data is around 10 % for the variables *Temperature* (T) and *Relative Humidity* (RH) and around 11 % for the *Rainfall Occurrence* (RO) variable.

For the T and RH data sets, the 10 % of missing data was approximated using the called Kriging interpolation method[§], a Gaussian process regression governed by prior covariances.

Since the *Rainfall Occurrence* is a binary variable (*i.e.* takes values of 1 or 0, for the occurrence or not of rainfall, respectively), the missing data in the RO data set cannot be obtained by the same interpolation method previously mentioned. Thus, the nearest-neighbour interpolation method (also known as proximal interpolation) was used instead. This method simply locate the nearest data value

relationship between the water demand and the summer months.

[§]In several cases, missing data is replaced by average values, however, interpolation methods can provide better approximations.

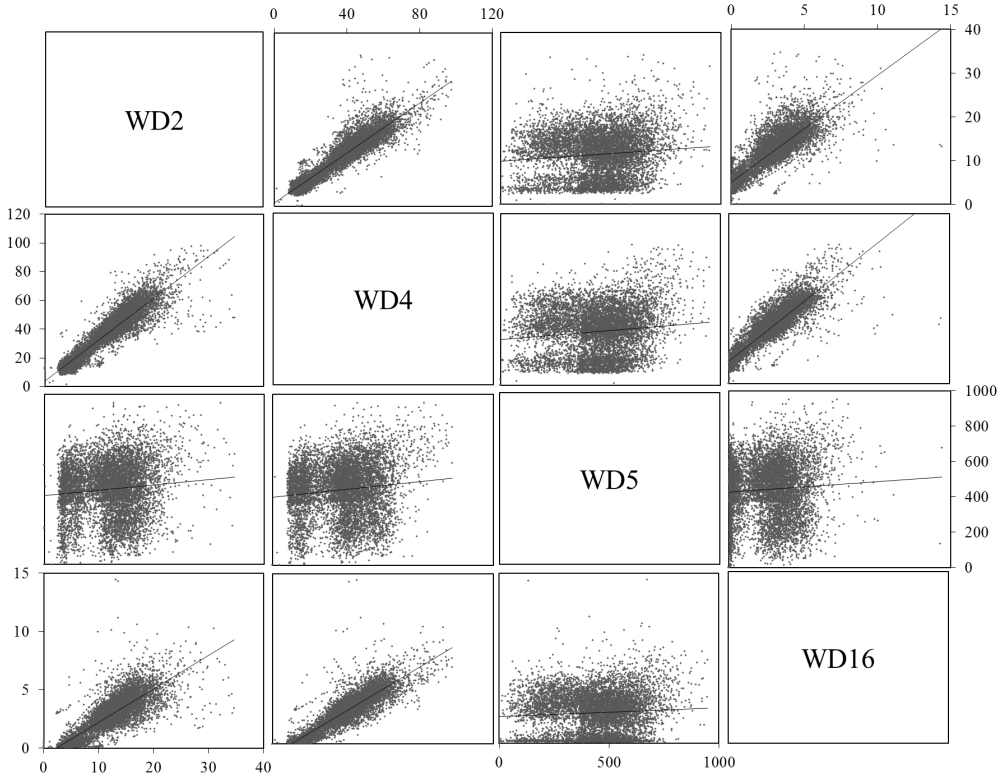


Figure 14.6: Scatter plot matrix showing the relationships between the water demand data sets that represent the water demand in neighbouring delivery points.

and assign the same value.

Both interpolation methods were implemented using the XonGrid interpolation Add-in[¶] for Excel.

Some statistical information such as minimum, maximum, average and standard deviation values for the three meteorological data sets is provided in Table 14.3.

Table 14.3: Statistical information about the meteorological data sets, including *Temperature* (T), *Rainfall Occurrence* (RO) and *Relative Humidity* (RH) (from 3/12/2012 to 31/07/2013).

	T (° C)	RO	RH
Total observations	5761	5761	5761
Minimum	-1.00	0.00	0.22
Maximum	35.00	1.00	1.05
Average	13.63	0.09	0.80
Standard deviation	4.88	0.29	0.16

In order to perform a first analysis to the influence of the weather variables in the water demand, the Pearson correlation coefficients between all variables were computed and graphically represented in Figure 14.7. The first impression is that the relationship between such variables is weak, since the coefficients obtained were so low. At the same time, from the three distinct variables, *Temperature* and *Relative Humidity* appear to have the strongest correlation with water demand.

[¶] `http://xongrid.sourceforge.net/`

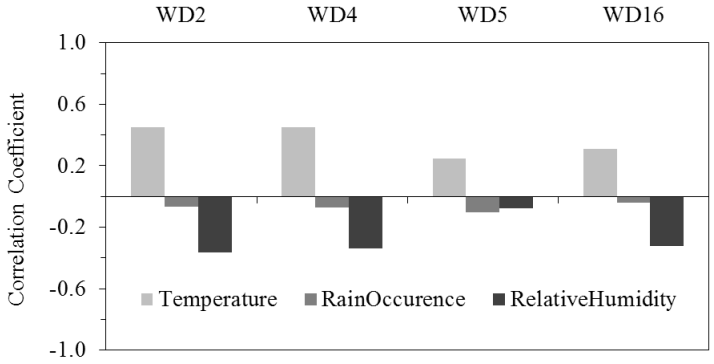


Figure 14.7: Pearson correlation coefficients between the water demand in each data set and the weather variables.

From the scatter plots provided in Figure 14.8, through the represented trend lines it is possible to observe the similar relationship (but in a symmetric way) of both *Temperature* and *Relative Humidity* with the water demands. The highest demands occur typically for higher temperatures and lower relative humidity. However, this is not so notorious for the WD5 data set.

The variable *Rainfall Occurrence*, that presented the lowest Pearson correlation coefficients, in a certain way demonstrates to cause some kind of influence in the water demands. In the scatter plots of 14.8, the highest demands occur typically in hours without rain occurrence (although again, this is not so notorious for WD5). For this reason, forecasting models including this variable will also be tested.

Besides the analysis to anthropic and weather variables, an analysis to the water demand time series lags was also performed. The idea is to verify which demands in previous hours present higher correlation with the current demands. Thus, the correlation coefficients between the current time series and the time series for lags 1 to 168 (previous one hour to one week) were computed. Results of these autocorrelation functions for each data set showing the more significant lags are provided in Figure 14.9.

For all data sets, the hours that demonstrate higher correlation with the current hour are the previous 1, 24 and 168 hours. However, while the highest correlation for the datasets WD2 and WD4 was obtained for the 168-hours lag, for the data sets WD5 and WD16, the 1-hour lag has an higher correlation. Thus, it was decided to test the three lags in the forecasting models.

14.4 Hourly forecasting models development

In order to identify the variables with more influence in the Portuguese network water demands and develop the most adequate model to predict the future demands in a hourly basis, four distinct water demand data sets were used and several forecasting models with distinct inputs were developed.

Firstly, the simplest forecasting models using only the current water demand series were developed, including naïve models, the classic Exponential Smoothing and ANN-based models. The intention of developing naïve models is essentially to be a reference. To be considered adequate, any other developed forecasting model is intended to present, at least, equal or superior accuracy to those ones.

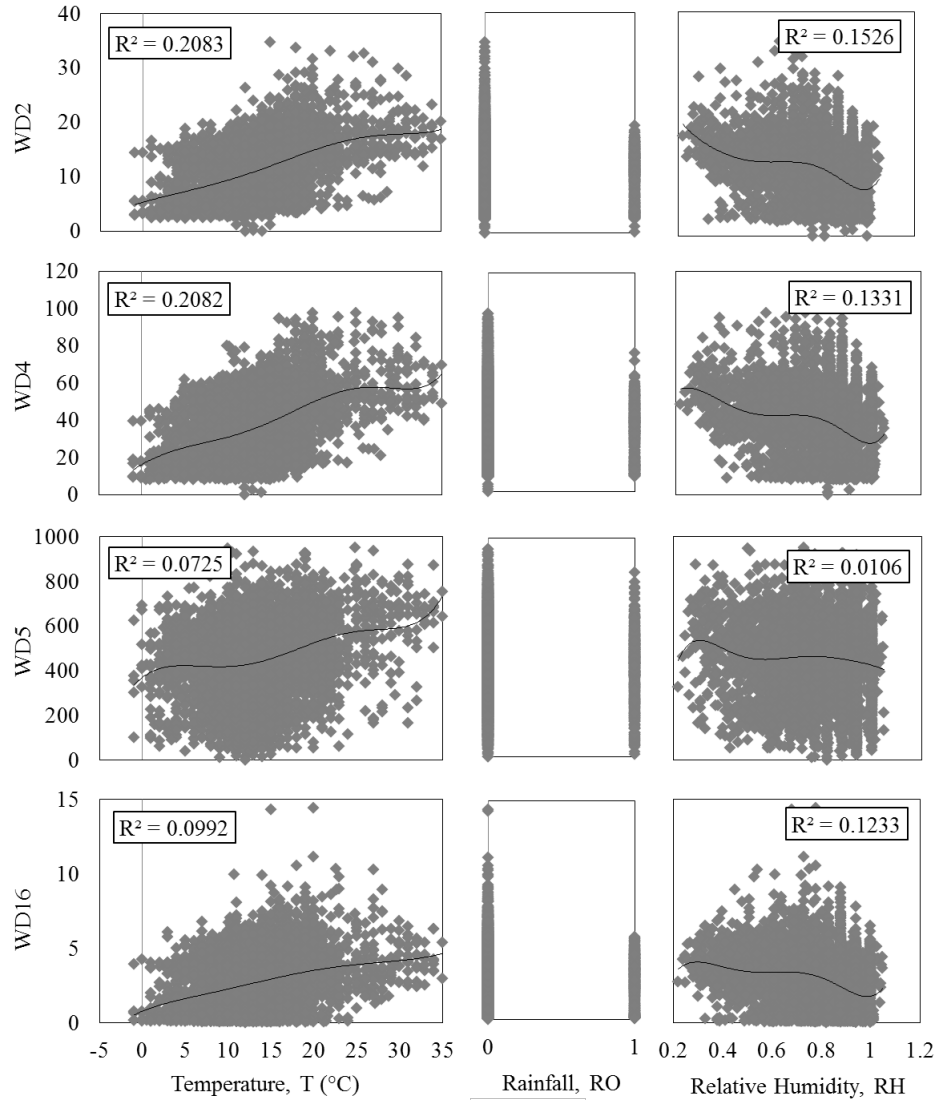


Figure 14.8: Scatter plots showing the relationship between the water demand (in m^3/h) and the weather variables *Temperature* (T), *Rainfall Occurrence* (RO) and *Relative Humidity* (RH). Adjusted 6^{th}-order polynomial trend lines and the corresponding squared correlation coefficients are also represented.

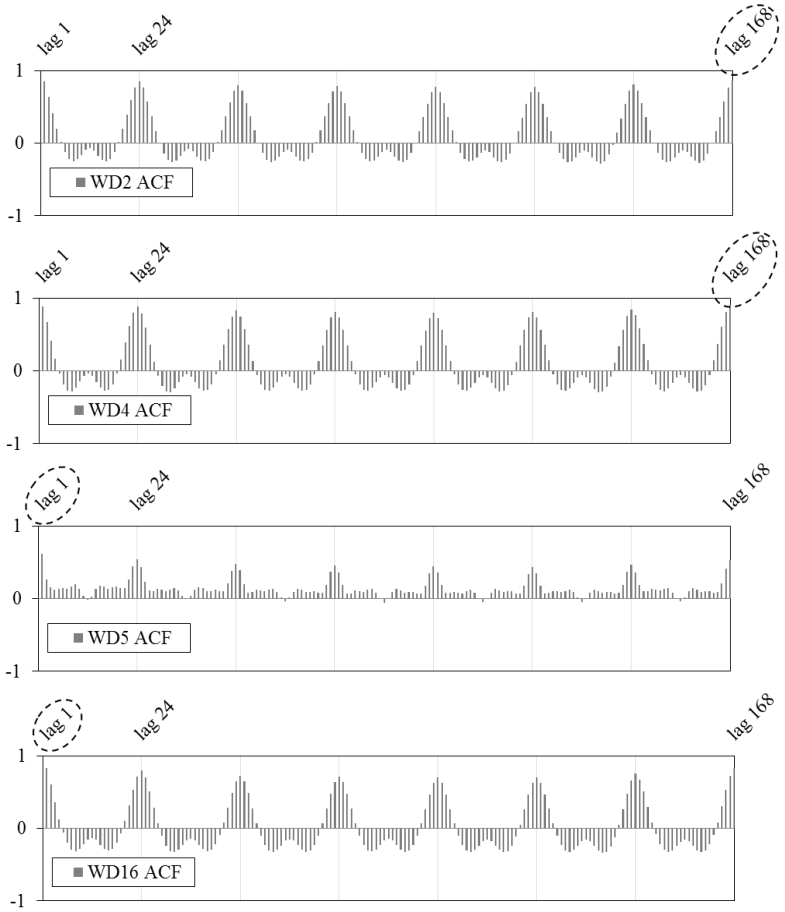


Figure 14.9: Autocorrelation Functions (ACF) for the distinct water demand time series considered in this work. The black dashed lines mark the lag that presents the highest correlation in each case.

For its simplicity of implementation, both the Naïve models and the Exponential Smoothing models were developed using Microsoft Excel.

Seasonal naïve models with a seasonality of one week (168 hours) were developed. Since the data sets demonstrated also high correlations with the 1-hour lagged series (especially the data sets WD5 and WD16), simple naïve models were also developed.

Concerning the Exponential Smoothing methods, both the Additive Seasonal Holt-Winters and the Multiplicative Seasonal Holt-Winters were developed for each data set. After that, distinct ANN-based models with additional input variables were developed. The idea is to test the influence of distinct variables in the models separately and then verify if the simultaneous use of the variables with more influence allow obtaining better forecasting models.

All ANN-based models were developed using Matlab R2012a, more specifically using the called narnet and narxnet models, for single and multiple predictors, respectively. Narnet, or non-linear autoregressive neural network, is used to predict a time series from that series past values (*MathWorks: narnet*, 2015). Narnet is a recurrent network (has feedback connections), where the feedback only depends on past outputs. It can be represented as narnet(FD,HN), where FD corresponds to the feedback delay and HN is a vector of the number of neurons in the hidden layer(s).

Narxnet, or non-linear autoregressive neural network with external input, is used to predict one time series given past values of the same time series (the feedback input) and another time series, called the external or exogenous time series (*MathWorks: narxnet*, 2015). Similarly with the narnet, a narxnet can be represented as narxnet(ID,FD,HN), where ID is the input delay of the external time series.

Table 14.4 provides a list of all ANN models developed to test the influence of distinct input variables.

Table 14.4: Input variables selected for each distinct ANN-based model developed in this work.

Data set	ANN model	Input variables	Data set	ANN model	Input variables
WD2	WD2_hist	WD2(t)	WD16	WD16_hist	WD16(t)
	WD2_1lag	WD2(t, t-168)		WD16_1lag	WD16(t, t-1)
	WD2_3lags	WD2(t, t-1, t-24, t-168)		WD16_3lags	WD16(t, t-1, t-24, t-168)
	WD2_anthrop	WD2(t), Hour, Month		WD16_anthrop	WD16(t), Hour
	WD2_neighb	WD2(t), WD4(t), WD16(t)		WD16_neighb	WD16(t), WD2(t), WD4(t)
	WD2_meteo	WD2(t), T(t), RH(t)		WD16_meteo	WD16(t), T(t), RH(t)
	WD2_rain	WD2(t), RO(t)		WD16_rain	WD16(t), RO(t)
	WD2_selection	selected variables		WD16_selection	selected variables
	WD2_all	all variables		WD16_all	all variables
WD4	WD4_hist	WD4(t)	WD5	WD5_hist	WD5(t)
	WD4_1lag	WD4(t, t-168)		WD5_1lag	WD5(t, t-1)
	WD4_3lags	WD4(t, t-1, t-24, t-168)		WD5_3lags	WD5(t, t-1, t-24, t-168)
	WD4_anthrop	WD4(t), Hour, Month		WD5_anthrop	WD5(t), Hour, Month
	WD4_neighb	WD4(t), WD2(t), WD16(t)		WD5_meteo	WD5(t), T(t)
	WD4_meteo	WD4(t), T(t), RH(t)		WD5_rain	WD5(t), RO(t)
	WD4_rain	WD4(t), RO(t)		WD5_selection	selected variables
	WD4_selection	selected variables		WD5_all	all variables
	WD4_all	all variables			

The first developed models (using narnet) receive as only input the historical data, considering the current data series, WD(t). After that, in each of the other models, different types of variables

(external inputs) are additionally included (using narxnet). The WD_1lag models consider as additional input the lagged series that presented the highest correlation coefficient (according to Figure 14.9). The WD_3lags models consider the addition of the three lagged series that presented the highest correlation coefficients (1, 24 and 168 hours for all data sets). The WD_anthrop models include the selected anthropic variables for each data set (according to Figure 14.5).

For all data sets except WD5, models considering as additional input the water demand series of the neighbouring areas were developed.

The meteorological variables that presented the higher correlation coefficients with each data set were included in the WD_meteo models. However, since it was verified that the variable *Rainfall Occurrence* could present some influence in the water demand (see Figure 14.8), a separate model to test the influence of including such variable was also developed for each data set (WD_rain).

Finally, for each data set, a model using as input the variables of the two models that presented the best forecast accuracies was develop (WD_selection) and compared with another model considering all variables as input (WD_all).

Forecasting results with the distinct methods for each data set are compared using scale-dependent accuracy measures. Scaled/normalised accuracy measures are also computed to allow the comparison between the distinct data sets.

Data sets division

For the development of the traditional forecasting models (naïve and exponential smoothing), each water demand data set was divided into two subsets: fitting and validation. The first 80 % of data (6036 observations) was used for fitting the model while the 20 % remainder data (the 1500 most recent observations) was left to validate the developed model.

Concerning the ANN-based forecasting models, the same amount of data (the 1500 most recent observations) was left for the final validation of each model. The remaining data (6036 observations) was used to develop the neural network: 70 % for training, 15 % for cross-validation and another 15 % for testing.

Since the meteorological data sets contain less 1775 observations than the water demand set (as already shown in Tables 14.1 and 14.3), the first 1775 observations of the water demand data sets were discarded, giving a total number of observations equal to 5761 for each data set, instead of 6036 (2867 data points for training, 613 for cross-validation and 613 for testing).

Neural networks architecture

A simple tool to select the most appropriate architecture for each ANN-based model was developed using Matlab. A single hidden layer network was considered for all cases, varying only the number of nodes. The developed tool automatically performs (i) the networks architecture selection, (ii) the networks development and (iii) forecast with new data (final validation). The main followed steps are:

1. Water Demand series autocorrelation function (ACF) and partialautocorrelation function (PACF) computation.
2. Definition of the number of input delays and feed-back delays: ID $=$ max(ACF) and FD $=$ max(PACF).
3. For 1 to 10 hidden nodes (HN), considering always the same random variables for the weights

initialisation:

(a) narnet(FD,HN) or narxnet(ID,FD,HN) generation;

(b) network training and test with target WD series feedback (open-loop network);

(c) open-loop network performance computation (Mean Squared Error).

4. Selection of the number of hidden nodes according to the best open-loop network performance obtained.
5. Close the network loop for forecasts without target feedback (only output feedback).
6. For 1 to 10 runs, considering always distinct random variables initialisation:
   (a) Predict missing values using the closed-loop trained network;
   (b) Compute the forecast accuracy (network performance) using the corresponding data for validation;
7. Save the network with best performance.

14.5 Forecasting results for the traditional methods

Following the data division previously mentioned, each traditional forecasting model was developed using the first data set (fitting subset). After that, hourly forecasts were obtained using the fitted models for the same period of the validation data set. The forecasting accuracy was computed for (i) the first hour predicted, (ii) the first 24 hours predicted and (iii) all the validation data set period (last 1500 observations - almost 9 weeks) predicted.

Tables 14.5 to 14.8 provide the forecasting accuracy measures of all traditional methods developed for each water demand data set.

Table 14.5: Fitting forecasting accuracy measures obtained for each data set with the Naïve, Seasonal Naïve, Additive Seasonal Holt-Winters and Multiplicative Seasonal Holt-Winters models.

Data set	Forecasting method	R^2 (-)	NSE (-)	MAE (m^3/h)	RMSE (m^3/h)	MAPE (%)	maxAE (m^3/h)
WD2	Naïve	0.73	0.71	1.90	2.74	1.67E+12	21.61
	Seas. Naïve	**0.79**	0.78	1.54	2.38	8.85E+12	20.35
	Add H-W	0.01	**1.00**	**0.01**	**0.01**	1.67E+10	**0.01**
	Mult H-W	0.01	**1.00**	**0.01**	**0.01**	9.43E+09	**0.01**
WD4	Naïve	0.78	0.77	5.54	7.46	1.19E+12	61.08
	Seas. Naïve	**0.88**	0.87	3.57	5.55	3.31E+12	42.83
	Add H-W	0.01	**1.00**	**0.00**	**0.01**	2.19E+09	**0.03**
	Mult H-W	0.01	**1.00**	**0.01**	**0.02**	7.49E+09	**0.09**
WD5	Naïve	**0.33**	0.15	106.29	139.87	5.61E+10	934.13
	Seas. Naïve	0.26	0.03	113.45	149.97	4.07E+12	709.43
	Add H-W	0.01	**1.00**	**0.09**	**0.09**	5.10E+08	**0.16**
	Mult H-W	0.01	**1.00**	**0.09**	**0.09**	4.99E+08	**0.19**
WD16	Naïve	0.71	0.68	0.69	0.98	6.21E+10	11.50
	Seas. Naïve	**0.74**	0.72	0.57	0.93	3.40E+11	12.05
	Add H-W	0.00	**1.00**	**0.00**	**0.00**	1.25E+08	**0.00**
	Mult H-W	0.00	**1.00**	**0.00**	**0.00**	4.71E+07	**0.00**

Table 14.6: First hour validation forecasting accuracy measures obtained for each data set with the Naïve, Seasonal Naïve, Additive Seasonal Holt-Winters and Multiplicative Seasonal Holt-Winters models.

Data set	Forecasting method	R^2 (-)	NSE (-)	MAE (m^3/h)	RMSE (m^3/h)	MAPE (%)	maxAE (m^3/h)
WD2	Naïve	_	_	4.18	4.18	35.86	4.18
	Seas. Naïve	_	_	**1.86**	**1.86**	**15.95**	**1.86**
	Add H-W	_	_	5.09	5.09	43.64	5.09
	Mult H-W	_	_	5.23	5.23	44.85	5.23
WD4	Naïve	_	_	3.99	3.99	9.21	3.99
	Seas. Naïve	_	_	12.07	12.07	27.84	12.07
	Add H-W	_	_	**0.86**	**0.86**	**1.98**	**0.86**
	Mult H-W	_	_	1.09	1.09	2.51	1.09
WD5	Naïve	_	_	**97.15**	**97.15**	**43.73**	**97.15**
	Seas. Naïve	_	_	111.92	111.92	50.38	111.92
	Add H-W	_	_	169.36	169.36	76.23	169.36
	Mult H-W	_	_	218.90	218.90	98.53	218.90
WD16	Naïve	_	_	0.45	0.45	18.84	0.45
	Seas. Naïve	_	_	2.65	2.65	110.23	2.65
	Add H-W	_	_	**0.35**	**0.35**	**14.62**	**0.35**
	Mult H-W	_	_	**0.35**	**0.35**	14.63	**0.35**

Table 14.7: First 24 hours validation forecasting accuracy measures obtained for each data set with the Naïve, Seasonal Naïve, Additive Seasonal Holt-Winters and Multiplicative Seasonal Holt-Winters models.

Data set	Forecasting method	R^2 (-)	NSE (-)	MAE (m^3/h)	RMSE (m^3/h)	MAPE (%)	maxAE (m^3/h)
WD2	Naïve	0.00	-1.07	5.02	6.66	94.01	12.41
	Seas. Naïve	**0.86**	**0.81**	**1.33**	**2.01**	**10.35**	**5.12**
	Add H-W	0.33	0.34	5.15	5.45	64.52	9.57
	Mult H-W	0.26	0.31	5.19	5.58	60.22	9.27
WD4	Naïve	0.00	-0.07	13.11	16.14	73.95	30.36
	Seas. Naïve	**0.80**	**0.70**	**6.08**	8.62	**16.49**	27.15
	Add H-W	0.00	-0.41	6.46	**7.98**	22.10	20.56
	Mult H-W	0.01	-0.34	6.50	**7.76**	24.69	**19.75**
WD5	Naïve	0.01	-0.89	110.35	129.40	28.80	292.61
	Seas. Naïve	0.33	**-0.32**	**96.86**	**108.10**	**26.47**	**201.05**
	Add H-W	0.54	-1195.96	206.95	232.19	60.18	361.13
	Mult H-W	**0.62**	-4119.31	395.20	430.80	108.25	643.13
WD16	Naïve	0.00	-0.04	1.40	1.68	418.31	2.75
	Seas. Naïve	**0.79**	0.70	**0.59**	**0.90**	**37.88**	2.65
	Add H-W	0.03	**0.98**	0.82	1.03	48.73	**2.38**
	Mult H-W	0.03	**0.98**	0.82	1.03	48.49	**2.38**

Table 14.8: All data validation forecasting accuracy measures obtained for each data set with the Naïve, Seasonal Naïve, Additive Seasonal Holt-Winters and Multiplicative Seasonal Holt-Winters models.

Data set	Forecasting method	R^2 (-)	NSE (-)	MAE (m^3/h)	RMSE (m^3/h)	MAPE (%)	maxAE (m^3/h)
WD2	Naïve	0.09	-0.10	5.03	6.41	69.09	17.85
	Seas. Naïve	**0.73**	0.73	**2.21**	**3.16**	**16.25**	**17.67**
	Add H-W	0.00	**1.00**	3.36	4.23	31.99	18.46
	Mult H-W	0.00	**1.00**	3.40	4.29	31.83	18.43
WD4	Naïve	0.10	0.07	16.75	20.24	54.41	58.35
	Seas. Naïve	**0.68**	0.68	9.33	11.92	21.52	**41.54**
	Add H-W	0.03	**1.00**	9.40	12.78	21.18	46.89
	Mult H-W	0.00	**1.00**	**8.66**	**11.87**	**20.23**	44.89
WD5	Naïve	0.31	-1.08	224.65	256.55	41.02	630.02
	Seas. Naïve	0.26	0.21	**124.55**	**157.84**	**26.03**	**517.76**
	Add H-W	**0.39**	**0.80**	159.38	194.80	38.82	598.67
	Mult H-W	0.48	0.24	324.53	376.91	72.09	996.23
WD16	Naïve	0.02	0.02	1.66	2.07	361.07	11.56
	Seas. Naïve	**0.55**	0.46	**0.86**	1.53	**35.18**	13.66
	Add H-W	0.04	**1.00**	1.05	1.51	47.07	12.71
	Mult H-W	0.00	**1.00**	1.04	**1.47**	51.30	**12.39**

Observing table 14.5, it is possible to see that the Exponential Smoothing methods are better in fitting the data than both the Naïve models for all data series. However, analysing the validation forecast accuracy for the first 24 hours (table 14.7), the exponential smoothing methods only perform better for the data series WD4 and WD16. For the WD2 data set, the Seasonal Naïve model revealed to be better than any of the other traditional methods. On the other hand, the Simple Naïve provided the best results for the WD5 data set, which presented higher correlation with the 1-lagged series than the 168-lagged series.

Results demonstrated that, for exponential smoothing methods, perfect fitting does not imply a good forecast accuracy. At the same time, comparing tables 14.6 and 14.7, it can be concluded that the method that best predict the first hour, may not be the best method to predict the first 24 hours.

The Seasonal Naïve model presented good performance when predicting 24-hours or even the (almost) 9 weeks ahead.

Given the analysed results, both the Seasonal Holt-Winters methods may not be the most appropriate to predict the water demands. This is probably because the serial dependence in the observations may not be appropriately captured by these approaches. For this reason, it is important to compare the results with distinct approaches capable of including other properties, such as the ANN-based methods.

14.6 Forecasting results for the ANN-based methods

Similarly with the procedure followed for the traditional forecasting methods, the validation accuracy measures for each ANN-based model were computed for (i) the first predicted hour, (ii) the first 24 hours predicted and (iii) the entire validation set dimension prediction. Detailed results for all tested

models can be found in appendix E. Table 14.9 provides the best ANN-based models results as well as the corresponding automatically selected architecture for the water demand forecast 24 hours in advance. Such results are compared with the ones obtained with the Seasonal Naïve. In Figure 14.10, the graphical representation of such results is also provided for a more complete interpretation.

Table 14.9: Results of the first 24 hours forecast accuracy measures for the best ANN-based models obtained in each data set compared to the Seasonal Naïve method.

Best models	Network architecture	R^2 (-)	NSE (-)	MAE (m^3/h)	RMSE (m^3/h)	MAPE (%)	maxAE (m^3/h)
WD2_1lag	narxnet(1:168,1:1,6)	0.77	0.77	1.72	2.16	16.16	**4.67**
WD2_anthrop	narxnet(1:168,1:1,3)	0.74	0.73	1.59	2.33	12.96	6.24
WD2 Seas. Naïve	_	0.86	0.81	1.33	2.01	10.35	5.12
WD4_1lag	narxnet(1:168,1:1,8)	0.88	0.87	3.91	5.49	11.33	12.96
WD4_neighb	narxnet(1:168,1:1,2)	**0.94**	**0.93**	**3.09**	**3.88**	**10.81**	**8.84**
WD4 Seas. Naïve	_	0.80	0.70	6.08	8.62	16.49	27.15
WD5_3lags	narxnet(1:1,1:1,8)	0.25	0.01	79.79	95.58	25.47	209.81
WD5_rain	narxnet(1:1,1:1,7)	0.08	0.00	68.78	95.96	23.36	282.76
WD5 Seas. Naïve	_	0.33	-0.32	96.86	108.10	26.47	201.05
WD16_3lags	narxnet(1:1,1:1,6)	0.40	0.39	1.00	1.23	275.25	2.70
WD16_anthrop	narxnet(1:1,1:1,5)	0.87	0.87	0.45	0.57	38.81	1.11
WD16_selection	narxnet(1:1,1:1,10)	**0.92**	**0.91**	**0.38**	**0.47**	**24.12**	**0.87**
WD16 Seas. Naïve	_	0.79	0.70	0.59	0.90	37.88	2.65

An interesting first observation from Table 14.9 is related with the distinct input variables that provided the best forecast results for each data set. Although the four tested data sets correspond to water demand from regions close to each other, the influence that distinct variables cause in the models performance is notorious.

Starting from the WD2 and WD4 time series, that presented the highest autocorrelation with the 168h-lagged series, both demonstrated better results when including such lagged series as model input. However, while the anthropic variables allowed to achieve one of the best results with ANN-based models for the WD2 series, in the case of the WD4 series this does not occur. In turn, for this last time series data set, the inclusion of historical water demands of neighbour sites (WD2(t) and WD16(t)) significantly improved the forecasting models performance. As represented in Figure 14.10, while the model for predicting the WD4 series including the 168-hour lag (WD4_1lag) presented predicted values quite below to the targets, the model considering the neighbourhood past demands (WD4_neighb) was able to provide a better fitting with the targets.

Observing the best ANN-based models obtained for WD5 and WD16, in both cases the inclusion of the 3 most correlated lagged series allow to improve the forecasting results (see Table 14.9). However, the other variables that also improved the series prediction are not coincident. Models to predict WD5 perform better when including the variable *Rainfall Occurrence*, while models to predict WD16 perform significantly better with the simultaneous use of the three more significant lagged series (WD16(t-1), WD16(t-168) and WD16(t-1)) and the anthropic variable *Hour* (*i.e.* the WD16_selection model). From Figure 14.10 it is observed that the model WD5_1lag is capable of detecting variations in demand while the model WD5_rain, despite resulting in slightly better accuracies, presents predicted values almost constant during the day (similar to the average of the observations). Concerning the graphs of the WD16 time series results, the WD16_anthrop and the

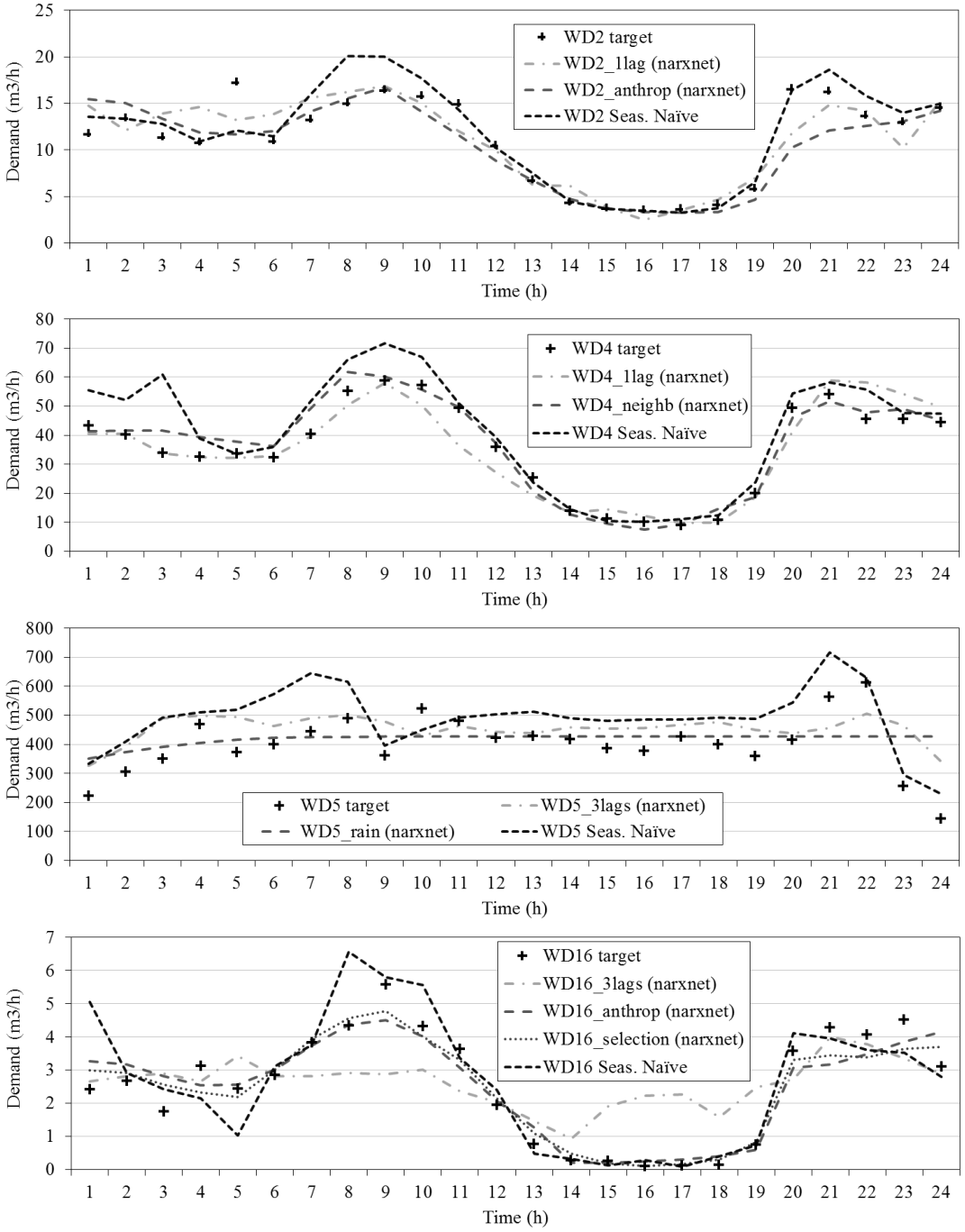


Figure 14.10: Graphical representation of the first 24 hours predictions of water demands with the models that provided the best results for each distinct dataset (same models presented in table 14.9) compared with the expected values (target).

WD16_selection models are clearly the ones presenting the best fitting with the targets.

The ANN-based models did not provided significantly good performances when compared with the seasonal naïve for predicting the WD2 and WD5 series. However, for the WD4 and WD16 series, the ANN-based models outperformed the traditional naïve.

It is important to mention that the use of all variables that apparently demonstrated to have influence on the water demands (from the preliminary correlation and scatter plots analysis) as model input, does not necessarily improve the forecasts performance. In fact, from the results presented in appendix E, it is possible to observe that, in almost all cases, the use of all variables as input decreased the forecast model performance when compared with the simple ANN-model using only the historical demands. This is possibly occurring due to the increase of the neural networks complexity.

15. Locating potential sites for energy recovery in an Italian case-study

An Italian network is used to test and validate the developed tool for energy recovery. A preliminary financial analysis to the hydraulic turbines design is provided. The results are compared with other authors.

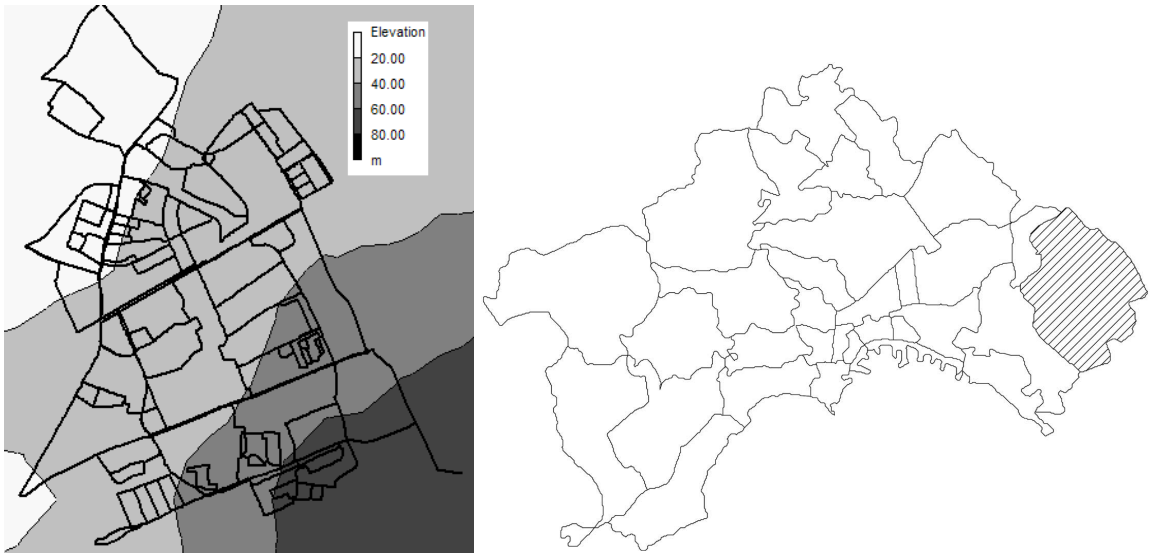
15.1 Case-study description

The network analysed in this chapter, the Napoli Est water distribution system, was introduced by Fontana et al. (2011)*, who tested the use of pressure reduction valves (PRVs) and pumps-as-turbines (PATs) for losses reduction and energy recovery. The Napoli Est network serves around 8 % of the Naples municipality (see the served area represented in Figure 15.1b), which corresponds to a number of inhabitants around 65000-70000. The elevation in the area ranges between 11-78 m above sea level (see the elevation contour plot in Figure 15.1a) (Fontana et al., 2011). The distribution network is composed of 349 pipes with diameters ranging from 40 to 1000 mm and 251 nodes, from which 151 are demand nodes, as represented in figure 15.2a. The network is supplied by the San Sebastiano reservoir whose pattern of water level variation for the period of simulation considered in this work (72 hours) is represented in the top graph of Figure 15.2b. Using the EPANET 2.0 for the simulation of the network, the maximum pressure obtained is observed in junction node J187 at 6 a.m. of each simulated day. The minimum pressure values are observed in junction nodes J182 and J183. As can be observed in the bottom graph of Figure 15.2b, the average value of pressure in the entire network is superior to 70 m, meaning that, considering the 25 m of minimum pressure required, the network presents excess of pressure.

This case study is used to present the potentialities of the developed tool. In order to validate the methodology and the implemented numerical tool, the results are compared with the ones presented by Fontana et al. (2011).

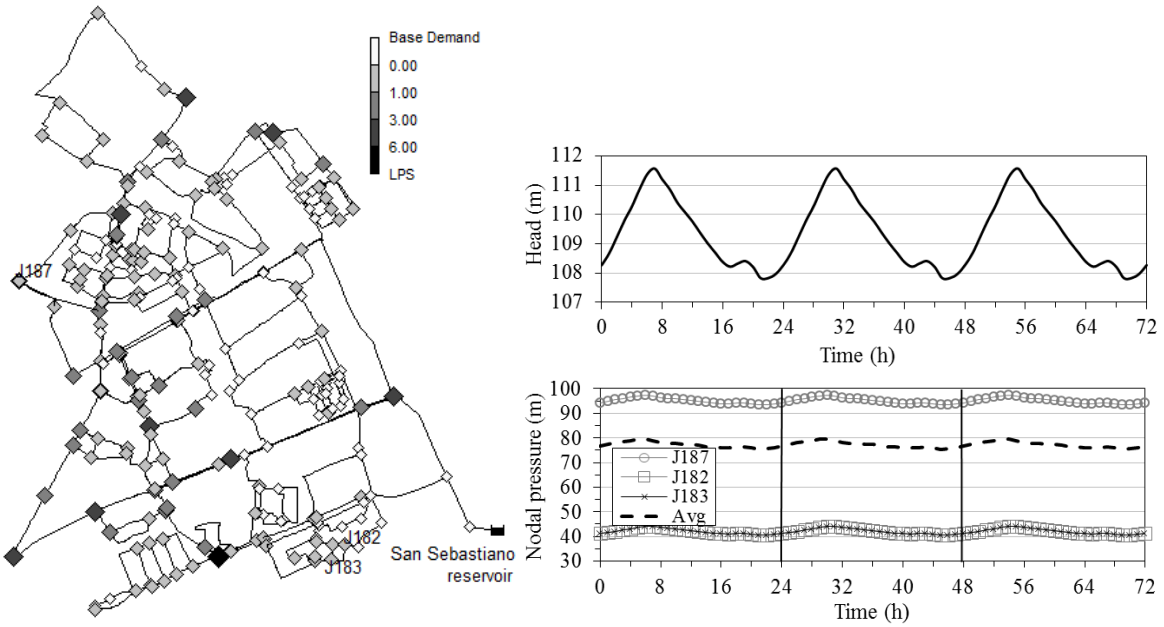
In their methodology, Fontana et al. (2011) defined 25 m as the minimum pressure for the demand nodes. However, they mention that pressures above 20 m can be accepted. The main objective of their

*The EPANET model of this network was gently provided by Professor Nicola Fontana from the Faculty of Engineering of the University of Sannio, Italy.



(a) (b)

Figure 15.1: Representation of the Italian case-study: (a) EPANET model of the Napoli Est network showing the differences in elevation and (b) location of the network in the Naples municipality (Fontana et al., 2011).



(a) (b)

Figure 15.2: Results of the Napoli Est network simulation: (a) values of base demand in the junction nodes and (b) pattern defined for the water level variation in the San Sebastiano reservoir (top graph) and pressure variation in the nodes with the maximum and minimum values identified, as well as the average pressure variation in all demand nodes of the network (bottom graph).

work was to find optimal locations for installing PRVs in order to minimise the water losses in the network while maintaining the required minimum pressure at the demand nodes. In the second stage, the objective was to find a possible replacement of such valves by PATs in order to recover the energy that is dissipated by the valves. However, the main objective of the methodology proposed in this work is to locate potential sites for energy recovery and suggest adequate hydraulic turbines for the maximum energy production, not constrained to PATs.

15.2 Testing the developed tool for energy recovery with hydroturbines

15.2.1 Locating and assessing potential sites for energy recovery

Following the method proposed in Chapter 10, at an initial stage, the hydraulic power available in each pipe of the network was computed. The best scenarios obtained, S1 to S8, are depicted in Table 15.1. The presented flow ranges correspond to the values obtained from the flow-duration curve drawn for each pipe and considering the values between the minimum and the mean flow rate. The available hydraulic power in each pipe was computed for all time steps of the 3-day simulation. The obtained results are also compared with the ones analysed by Fontana et al. (2011), represented by scenarios A, B and D.

Table 15.1: Results for the pipes of the Napoli Est network that presented the highest hydraulic power, S1 to S8, and for the pipes considered in the scenarios presented by Fontana et al. (2011), A, B and D.

Scenarios	Pipe ID	Flow range (l/s)	Head drop range (m)	Hydraulic power (kW)	Daily hydraulic power (kW/day)
S1	P325	28-124	5.433-13.637	88.40	29.47
S2	P262	6-44	3.651-12.773	31.40	10.47
S3	P354	239-339	0.062-0.120	30.49	10.16
S4	P29	9-13	1.091-2.267	22.99	7.66
S5	P266	8-13	7.892-19.666	19.93	6.64
S6	P358	9-26	0.077-0.144	18.37	6.12
S7	P261	8-16	3.594-12.572	16.65	5.55
S8	1	240-339	0.031-0.059	15.13	5.04
A	1	240-339	0.031-0.059	15.13	5.04
B	P151	143-204	0.044-0.086	13.27	4.42
	P200	62-86	0.009-0.018	1.15	0.38
D	P111	67-89	0.014-0.025	1.65	0.55
	P134	62-87	0.071-0.136	8.87	2.96
	P354	239-339	0.062-0.120	30.49	10.16
	P358	9-26	0.077-0.144	18.37	6.12

As can be observed, from the 8 best obtained results, three of the pipes correspond to the same pipes identified by Fontana et al. (2011) as locations with higher potential for water losses reduction and energy production. However, when taking only into account the energy production, other pipes reveal substantially higher potential, such as the scenarios S1 and S2. The locations of the pipes that correspond to the 8 best identified scenarios are represented in Figure 15.3 that also provides information about the pipes diameters.

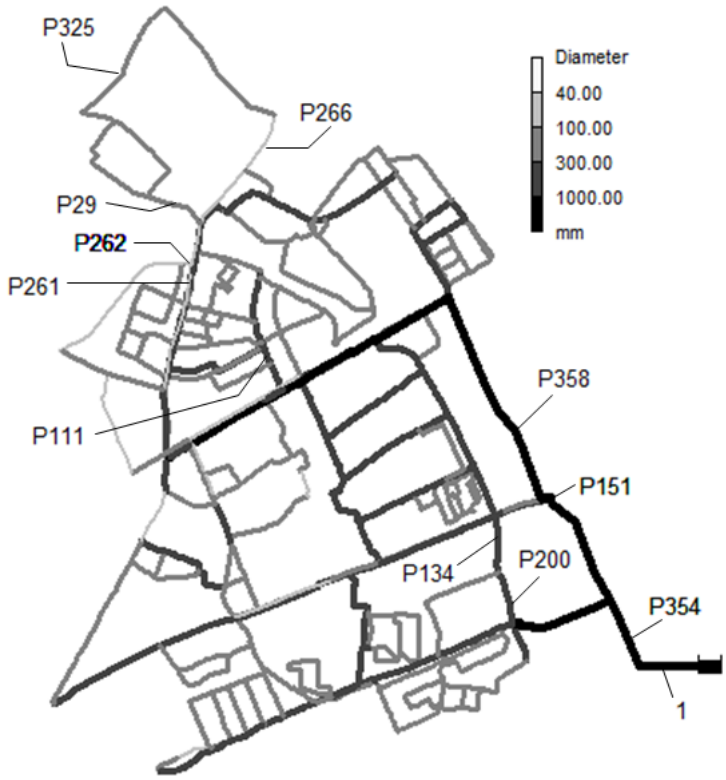


Figure 15.3: Location and diameters of the sites that represent the largest potential for energy recovery and also of the sites discussed in the work of Fontana et al. (2011).

The second stage of this analysis consists in the installation of virtual PRVs in the identified potential locations in order to obtain an approximation of the amount of potentially recoverable energy. During such procedure, the selection of the most profitable head loss coefficient for each valve was performed considering a minimum pressure limit of 25 m in the demand nodes. Table 15.2 shows the results for the selected scenarios obtained for each installed valve.

As can be observed, the implemented methodology to obtain an adequate head loss coefficient for the valves by maximising the energy production while maintaining the minimum required pressures in the network (10.8) provided results that differ from the ones presented by Fontana et al. (2011). In fact, for the common locations (*i.e.* in pipes P354, P358 and 1), the potentially recovered energy demonstrated to be superior to the expected by Fontana et al. (2011). However, it should not be forget that their main objective were the water losses reduction and not the energy production, contrarily to this work.

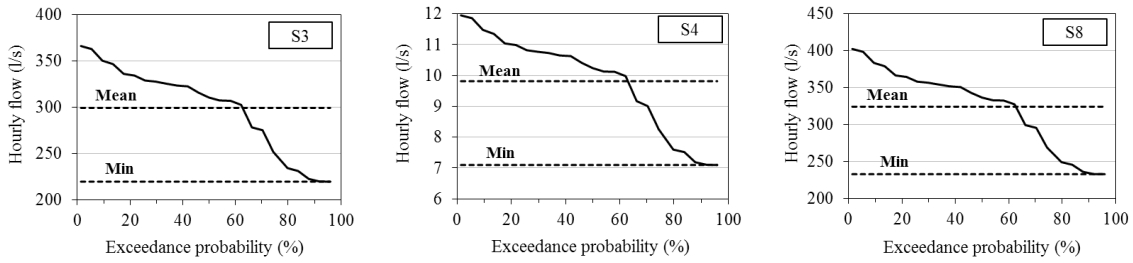
Results also demonstrated that the pipes that present higher combinations of flow rate and head drop do not necessarily present the highest potential for energy recovery. This is explained by the distinct effects on the network pressure drop caused by the installation of devices such as valves or turbines, demonstrating the importance of performing the simulation of the network operation accounting with such devices.

The sites that presented the largest values for the potentially recoverable power were pipes 1, P354 and P29. While the valves located in pipes 1 and P354 present high values of flow rate, the valve in pipe P29 presents potential essentially due to the values of head. The flow-duration curves (FDC) for

Table 15.2: Results for the energy dissipated (potentially recoverable energy) in the pressure reduction valves (PRVs) installed in the locations of the selected scenarios. The results of the reproduced scenarios proposed by Fontana et al. (2011) are also presented.

Scenarios	Pipe ID	PRV minor loss coeff., K (-)	Flow range, min-mean (l/s)	Head loss, min-mean (m)	Potentially recoverable power (kW/day)	
S1	P325	1500	4.03-6.11	20.09-47.95	74.56	
S2	P262	1000	0.56-1.01	10.16-35.26	9.47	
S3	P354	3000	219.6-299.4	11.94-22.73	1694.28	
S4	P29	3000	7.10-9.82	24.69-48.31	118.26	
S5	P266	1000	0.44-0.69	6.36-16.20	2.89	
S6	P358	1000	53.70-73.69	0.48-0.92	16.90	
S7	P261	500	0.48-0.89	3.76-13.56	3.21	
S8	1	3000	219.8-299.5	11.97-22.75	1696.50	
A	1	1000	232.8-324.2	0.30-0.41	723.35	723.35
B	P151	94000	89.19-123.49	6.41-12.62	389.47	530.38
	P200	4000	29.91-41.37	6.95-13.63	140.91	
D	P111	222000	3.18-4.01	1.43-2.30	2.26	817.76
	P134	114000	10.24-13.72	7.62-13.96	47.48	
	P354	1000	222.2-305.3	4.08-7.89	600.94	
	P358	73000	35.71-47.83	7.69-14.08	167.07	

these three sites[†], presented in Figure 15.4, illustrate the usable flow range for energy generation. For the three cases, it can be observed that during more than 60 % of the time (from the 73 hours) the hourly flow is equal to the mean (the value to be used as design flow for the turbine).



(a) Scenario S3 (b) Scenario S4 (c) Scenario S8

Figure 15.4: Flow-duration curves (FDC) for the three sites that presented the largest potential for energy recovery without compromising the pressure requirements: (a) S3, (b) S4 and (c) S8.

15.2.2 Selecting the most adequate turbines and testing possible choices

The first step in the selection of turbines for a specific site consists in the selection of the types of turbines that operate in the range of available head. According to that, the results obtained for the possible type of turbine to install in the eight selected scenarios are presented in Table 15.3.

As can be observed, scenario S6 (located in pipe P358) presents a very limited and low head range

[†]It should be noticed that the presented flow-duration curves were adapted to the amount of existing data. Thus, instead of a typical (and recommended) daily FDC obtained with years of recorded data, an hourly FDC obtained with the available 73 hours (3 days) of data is presented.

Table 15.3: Results for the types of turbine that can be installed in the sites of the selected scenarios according to the available head ranges.

Scenarios	Location (pipe ID)	Head range (m)	Selected types of turbine
S1	P325	20.09-47.95	Francis, Cross-flow, Kaplan/Propeller, PAT
S2	P262	10.16-35.26	Francis, Cross-flow, Kaplan/Propeller, PAT
S3	P354	11.94-22.73	Francis, Cross-flow, Kaplan/Propeller, PAT
S4	P29	24.69-48.31	Francis, Cross-flow, Kaplan/Propeller, PAT
S5	P266	6.36-16.20	Cross-flow, Kaplan/Propeller, PAT
S6	P358	0.48-0.92	PAT
S7	P261	3.76-13.56	Cross-flow, Kaplan/Propeller, PAT
S8	1	11.97-22.75	Francis, Cross-flow, Kaplan/Propeller, PAT

where the turbines typically do not operate. For this reason, only a specific type of pump-as-turbine could eventually be used in such site. In fact, Aline, Vincent, and Petras (2012) mention that PATS are the most adequate for values of installed power inferior to 30 kW, which means that, according to the available hydraulic power (also listed in Table 15.2), a PAT would be the most adequate type of turbine for scenarios S2 (9.47 kW), S5 (2.89 kW), S6 (16.90 kW) and S7 (3.21 kW).

Pelton and Turgo-type turbines were automatically excluded from the presented scenarios due to the typical higher head range of operation of these types of turbines.

Following the implemented method of computation, the results concerning the adequate specific speed, N_Q (metric units), the approximated runner diameter, D_a, and efficiency for each type of turbine, η_{turb}, according to the design flow, Q_d, and head, H_d, are provided in Table 15.4. The daily net energy potentially recoverable, E_{net}, considering the obtained values of efficiency for each distinct type of turbine are also presented.

Table 15.4: Characteristics obtained for each type of turbine according to the design flow and head specified for each scenario (excluding the scenarios adequate for PATs) and the resulting daily net energy that can be recovered.

Scen.	Location (pipe ID)	Q_d (m^3/s)	H_d (m)	E_{gross} (kWh/day)	Turbine type	N_Q (-)	d (m)	η_{turb} (-)	E_{net} (kWh/day)
S1	P325	0.006	47.95	74.56	Francis	86.6	0.041	0.813	60.62
					Kaplan/Propeller	115.5	0.041	0.841	62.70
					Cross-flow	_	_	0.790	58.90
S3	P354	0.3	22.73	1694.28	Francis	125.8	0.260	0.790	1338.57
					Kaplan/Propeller	167.8	0.260	0.894	1514.48
					Cross-flow	_	_	0.790	1338.48
S4	P29	0.009	48.31	118.26	Francis	86.3	0.050	0.819	96.81
					Kaplan/Propeller	115.1	0.050	0.846	100.10
					Cross-flow	_	_	0.790	93.43
S8	1	0.3	22.75	1696.50	Francis	125.8	0.260	0.790	1340.54
					Kaplan/Propeller	167.7	0.260	0.894	1516.47
					Cross-flow	_	_	0.790	1340.24

Comparing the results in terms of the turbines efficiency, and hence, the potentially net recovered energy, the Kaplan/Propeller-type turbines demonstrated to be slightly more profitable in all the

scenarios due to the higher efficiency values. However, when selecting a turbine, the range of flows operation should also be taken into account since distinct turbine types present distinct variations in the efficiency when operating at flow rates inferior to the design flow, as demonstrated in Figure 10.6 in the III Part. An adjustable Kaplan, for instance, demonstrates a more satisfactory operation (higher efficiencies) over a wide range of flow variations when compared with a propeller with fixed guide vanes and blades. This means that an adjustable Kaplan turbine would be more adequate for energy recovery in water supply and distribution networks.

Since the values of possible recoverable energy are obtained from results of the network simulation considering the behaviour of a PRV, which correspond in fact to approximated values, it is important to understand the differences when modelling a turbine using real curves. For this reason, an analysis was performed in the three best obtained scenarios, S3, S4 and S8, using data of turbines existent in the market.

The hydraulic curves of turbines (head vs flow) are usually not easy to find in the literature, not even in the manufacturers websites or catalogues. However, examples of real curves from Cornell and Mavel, for pumps-as-turbines, and a Kaplan-type turbine, respectively, were used. The selected reverse pumps were the Cornell 10TR1 and the Cornell 1 1/4TR1 (Cornell pump, 2015) and the Kaplan-type was the Mavel TM3_18″ (*Mavel: TM Micro Turbines*, 2015).

Figure 15.5 provides the headloss curves of each mentioned type of turbine. The turbines operations were simulated in EPANET by replacing, in each scenario, the existent PRVs by a general purpose valve (GPV) with an associated headloss curve in accordance with the turbine hydraulic curve that provides the recovered head for a certain flow rate.

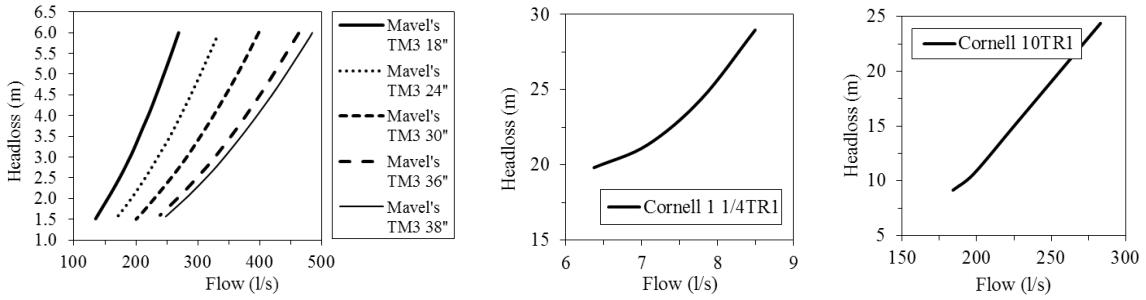


Figure 15.5: Headloss curves for the simulation of different types of turbines: (a) Mavel's Kaplan-type microturbines and (b) Cornell's pumps-as-turbines for lower and (c) for higher flow ranges.

Additionally, the curves presented by Fontana et al. (2011) for two reverse pumps (NC100-200 and NC150-200, whose hydraulic curves are represented in Figure 15.6), were also used and the results were compared with the ones obtained with the Cornell and Mavel's turbines.

From the existent data, different turbines were selected for each scenario, according to the flow and head characteristics of the sites. Table 15.5 presents the turbines tested in each selected scenario and the corresponding results in terms of energy recovered. The results for the scenarios proposed by Fontana et al. (2011) were also reproduced and are presented for a comparative analysis.

The rotational speed is 1550 rpm for the NC100-200 and NC150-200 PATs and 1200 rpm for the Cornell's PATs. The power output is 5 to 20 kW for the Mavel's TM3 micro turbines (*Mavel: TM Micro Turbines*, 2015) and 1 to 100 kW for the Cornell's PATs (Cornell pump, 2015). For the

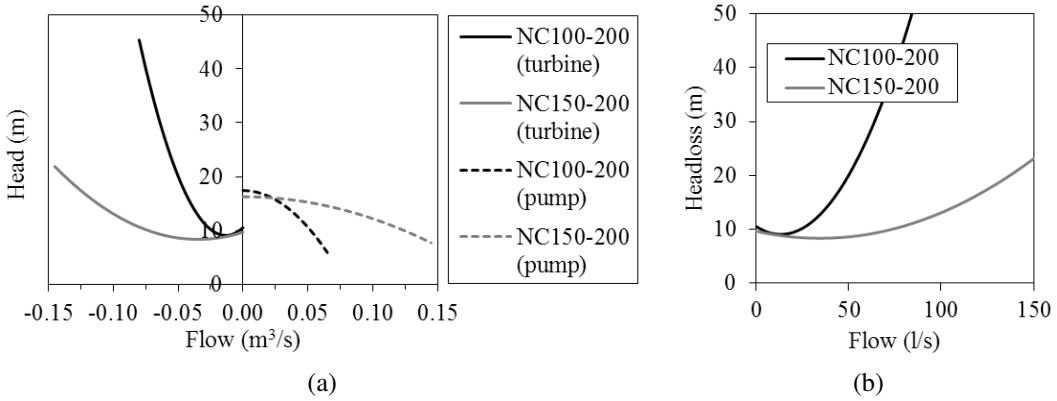


Figure 15.6: Representation of (a) the hydraulic curves of the two reverse pumps used by Fontana et al. (2011) and (b) the corresponding headloss curves used for the PATs simulation in EPANET.

turbines efficiencies, a value of 70 % was used for the PATs, while, for the Kaplan turbine, the value of efficiency obtained with the developed tool for this type of turbine in each scenario was used (Table 15.4).

Table 15.5: Results obtained from the simulation of the network operation considering the installation of different turbines available in the market. Results obtained from the reproduction of Fontana et al. (2011) scenarios are also presented.

Scen.	Location (pipe ID)	Simulated turbines	E_{gross} (kWh/day)	η_{turb} (%)	E_{net} (kWh/day)		Pressures <25 m?
S3	P354	Cornell 10TR1	1896.480	70.00	1327.54		Yes
		Mavel TM3_18″	679.047	89.40	607.07		No
		3 parallel NC150-200	1098.257	70.00	768.78		No
S4	P29	Cornell 1 1/4TR1	106.272	70.00	74.39		Yes
		NC100-200	27.755	70.00	19.43		No
S8	1	Cornell 10TR1	1898.344	70.00	1328.84		Yes
		Mavel TM3_18″	680.498	89.40	608.37		No
A	1	3 parallel NC150-200	991.12	70.30	773.31	773.31	No
B	P151	NC150-200	428.54	64.70	277.27	380.40	No
	P200	NC100-200	148.61	69.40	103.14		
D	P111	_	_	_	_	666.58	No
	P134	_	_	_	_		
	P354	3 parallel NC150-200	952.728	67.30	641.19		
	P358	NC100-200	49.220	51.60	25.40		

As can be observed, in the three scenarios where Cornell's PATs are tested, the daily energy recovered is larger. This is because, for the same range of flows, the head recovered by these turbines is superior when compared with the other turbines. However, the operation of the Napoli Est network with these specific PATs does not allow maintaining the required minimum pressures in the demand nodes. In fact, in some nodes, pressures inferior to 20 m were identified.

For scenario S3, the use of only one Kaplan-type turbine (Mavel TM3_18″) allowed to recover almost the same amount of daily energy when compared with the use of three parallel PATs (NC150-

200), while maintaining the required pressures in the network. It can also be observed that, for the same location (pipe P354), different scenarios produce different results. In the scenario proposed by Fontana et al. (2011) (scenario D), the same three parallel turbines installed in the same location were not capable of recovering the same amount of energy because the network was under distinct operational conditions: an additional PAT operating in other location of the network (pipe P358) and also two PRVs (at pipes P111 and P134).

Results demonstrate that, even performing a preliminary analysis on the potentially recovered energy in the networks, the real energy recovered will be highly dependent on the turbines selection from the range available in the market. Only a final simulation of the network operation with the information of a specific turbine can provide more precise results. However, the applied methodologies demonstrate quite good results for a preliminary analysis and serves as a good support for the the turbines selection.

15.2.3 Preliminary financial analysis

An important factor in the decision of the best turbine-type option for a specific scheme is the cost of the turbine and the corresponding cost of implementation. In the previous section, Kaplan turbines and PATs demonstrate to be adequate choices concerning their characteristics and suitability to the selected sites. In this section, a preliminary financial analysis focused in the turbines and respective installation costs is presented.

Results obtained from the developed tool for the distinct types of turbines are presented in Table 15.6. Despite the PATs costs computation are not included in the developed tool, the price ranges provided by Fecarotta, Aricò, Carravetta, Martino, and Ramos (2015) are used as a reference for comparison. For the installation costs, 15 % of the PATs cost was also considered. The energy tariff considered was the Italian tariff that is defined as 0.220 €/kWh (Fontana et al., 2011). It should be noticed that the presented simplified financial analysis, as a preliminary analysis, does not include, for instance, the annual maintenance costs, the cost of operations, nor the turbines life span. While the maintenance and operations costs may be similar for each analysed case, the turbines life span may cause influence in the decisions.

In a general overview, it becomes clear that a cross-flow turbines, besides presenting lower efficiencies when compared to other type of turbines, presents the lowest values of investment and the fastest return on investment (2 to 3 years in all scenarios). Only a PAT can be comparable to this type of turbine in economic terms. On the other hand, for the scenarios with the highest annual energy production, such as S3 and S8, the payback time does not differ significantly between the different types of turbines, except for PATs, which present a return on investment in one year instead of two. However, the life span of this type of turbine is usually lower when compared to the others.

A cash-flow analysis during the first 15 years for each scenario is presented in Figure 15.7. As already mentioned by Fecarotta et al. (2015), it should be remembered that the results obtained from this analysis depend on the considered tariff. For other countries, under distinct conditions of electricity selling price and possible incentives, the results can be different.

Although for scenarios S3 and S8, the Kaplan and fixed-propeller turbines present higher costs of investment, they also provide the highest long-term profits (more than 1500 k€ in 15 years).

Table 15.6: Results of the preliminary cost analysis for the selected scenarios.

Scen.	Turbine type	Energy (kWh/day)	Energy (MWh/year)	Revenue (€/year)	Turbine cost (€)	Installation cost (€)	Civil works (€)	Payback (years)
S1	Francis	60.62	22.13	4868	7249	1087	4535	3
	Kaplan	62.70	22.88	5034	8579	1287	4535	3
	Propeller	62.70	22.88	5034	6029	904	4535	3
	Cross-flow	58.90	21.50	4730	1576	236	4535	2
	PAT	52.19[a]	19.05	4191	1200-6800	180-1020	4535	2 to 3
S3	Francis	1338.57	488.58	107487	99804	14971	73054	2
	Kaplan	1514.48	552.79	121613	113507	17026	73054	2
	Propeller	1514.48	552.79	121613	80939	12141	73054	2
	Cross-flow	1338.48	488.54	107480	39456	5918	73054	2
	PAT	1185.99	432.89	95235	1200-6800	180-1020	73054	1
S4	Francis	96.81	35.34	7774	9610	1441	6351	3
	Kaplan	100.10	36.54	8038	11378	1707	6351	3
	Propeller	100.10	36.54	8038	7994	1199	6351	2
	Cross-flow	93.43	34.10	7502	2287	343	6351	2
	PAT	82.78	30.22	6648	1200-6800	180-1020	6351	2 to 3
S8	Francis	1340.54	489.30	107645	99804	14971	73091	2
	Kaplan	1516.47	553.51	121772	113513	17027	73091	2
	Propeller	1516.47	553.51	121772	80941	12141	73091	2
	Cross-flow	1340.24	489.19	107621	39471	5921	73091	2
	PAT	1187.55	433.46	95360	1200-6800	180-1020	73091	1

[a]Considering an efficiency of 70 % for all PATs.

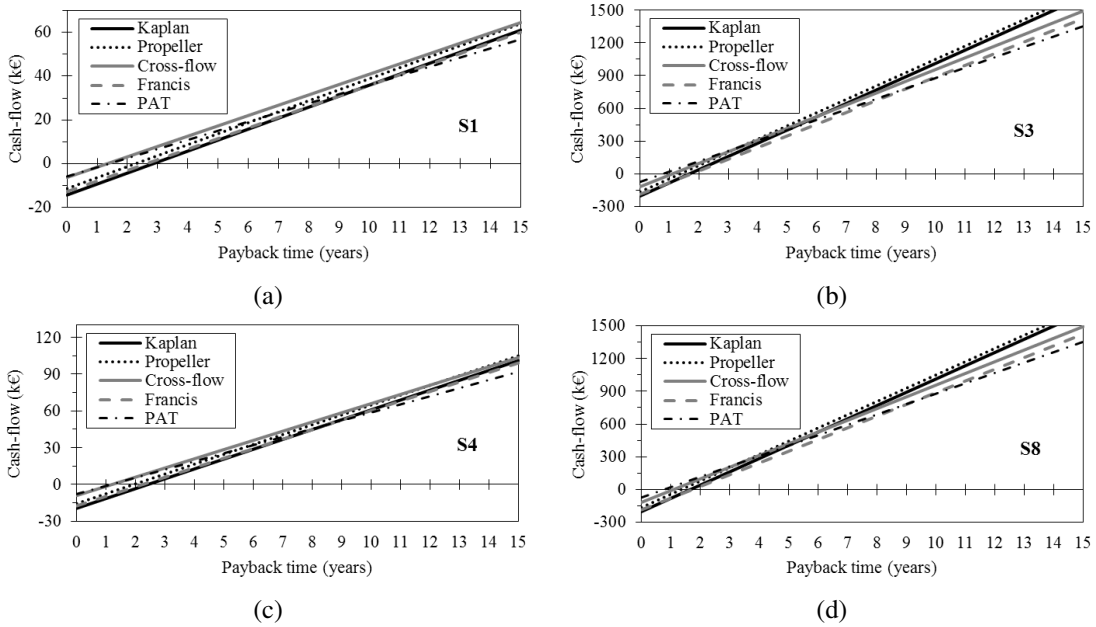


Figure 15.7: Cash-flow analysis for each selected scenario over a period of 15 years: (a) S1, (b) S3, (c) S4 and (d) S8.

References

Aline, C., Vincent, D., & Petras, P. (2012). Integration of small hydro turbines into existing water infrastructures. In H. Samadi-Boroujeni (Ed.), *Hydropower - practice and application* (chap. 12). Retrieved 2015, from http://www.intechopen.com/books/hydropower-practice-and-application/integration-of-small-turbines-into-water-infrastructure doi: 10.5772/35251

Carravetta, A., Fecarotta, O., Martino, R., & Antipodi, L. (2014). PAT efficiency variation with design parameters. *Procedia Engineering*, *70*, 285–291.

Chen, S., Montgomery, J., & Bolufé-Röhler, A. (2015). Measuring the curse of dimensionality and its effects on particle swarm optimization and differential evolution. *Applied Intelligence*, *42*(3), 514–526.

Coelho, B., & Andrade-Campos, A. (2014a). *On the comparison of numerical methodologies for control optimisation of variable-speed pumps.* Presented at IWA World Congress & Exhibition, 21-26September, Lisbon, Portugal. (Poster presentation)

Coelho, B., & Andrade-Campos, A. (2014b). *The operational costs minimisation in water supply systems using cascade optimisation techniques.* Presented at 20th Conference of IFORS, July, 13-18, Barcelona, Spain. (Oral presentation)

Coelho, B., & Andrade-Campos, A. (2016a). A new approach for the prediction of speed-adjusted pump efficiency curves. *Journal of Hydraulic Research*. (accepted for publication)

Coelho, B., & Andrade-Campos, A. (2016b). Numerical tool for hydraulic modelling – An educational approach. *International Journal of Mechanical Engineering Education*. (accepted for publication)

Cornell pump. (2015). *Cornell pump - hydro turbine.* http://www.cornellpump.com/products/hydroturbine.html.

Fecarotta, O., Aricò, C., Carravetta, A., Martino, R., & Ramos, H. M. (2015). Hydropower potential in water distribution networks: Pressure control by pats. *Water Resources Management*, *29*(3), 699–714.

Fontana, N., Giugni, M., & Portolano, D. (2011). Losses reduction and energy production in water distribution networks. *Journal of Water Resources Planning and Management*, *138*(3), 237–244.

GRIDS: Water GRIDS. (2015). http://grids.web.ua.pt/index.php/rd/dimeo/water-grids/.

Hamby, D. (1994). A review of techniques for parameter sensitivity analysis of environmental models. *Environmental monitoring and assessment*, *32*(2), 135–154.

Hashemi, S. S., Tabesh, M., & Ataeekia, B. (2014). Ant-colony optimization of pumping schedule

to minimize the energy cost using variable-speed pumps in water distribution networks. *Urban Water Journal*, *11*(5), 335–347.

Ingber, L. (1996). Adaptive simulated annealing (ASA): lessons learned. *Control and Cybernetics*, *25*(1), 33-54. Retrieved from `https://www.ingber.com/asa96_lessons.pdf`

Ingber, L. (2012). *Adaptive simulated annealing* (J. H.A. Oliveira, A. Petraglia, L. Ingber, M. Machado, & M. Petraglia, Eds.). New York: Springer. Retrieved from `https://www.ingber.com/asa11_options.pdf`

Kachitvichyanukul, V. (2012). Comparison of three evolutionary algorithms: GA, PSO, and DE. *Industrial Engineering and Management Systems*, *11*(3), 215–223.

López-Ibáñez, M., Prasad, T. D., & Paechter, B. (2011). Representations and evolutionary operators for the scheduling of pump operations in water distribution networks. *Evolutionary computation*, *19*(3), 429–467.

Marchi, A., Dandy, G., Wilkins, A., & Rohrlach, H. (2012). Methodology for comparing evolutionary algorithms for optimization of water distribution systems. *Journal of water resources planning and management*, *140*(1), 22–31.

Marchi, A., & Simpson, A. R. (2013). Correction of the epanet inaccuracy in computing the efficiency of variable speed pumps. *Journal of Water Resources Planning and Management*, *139*(4), 456–459.

Marchi, A., Simpson, A. R., & Ertugrul, N. (2012). Assessing variable speed pump efficiency in water distribution systems. *Drinking Water Engineering and Science Discussions*, *5*(1), 47–65.

Mathworks: narnet. (2015). `http://www.mathworks.com/help/nnet/ref/narnet.html`. (Accessed: 2015)

Mathworks: narxnet. (2015). `http://www.mathworks.com/help/nnet/ref/narxnet.html`. (Accessed: 2015)

Mavel: Tm micro turbines. (2015). `http://www.mavel.cz/turbines/tm-micro-turbines`.

Morton, W. R. (1975). Economics of ac adjustable speed drives on pumps. *IEEE Transactions on Industry Applications*(3), 282–286.

Natrella, M. (2010). *Engineering statistics handbook*. NIST/SEMATECH. Retrieved 2015, from `http://www.itl.nist.gov/div898/handbook/`

Pedersen, M. E. H. (2010). Good parameters for differential evolution. *Magnus Erik Hvass Pedersen*.

Quintela, A. C. (1981). *Hidráulica* (2nd ed.). Lisboa: Fundação Calouste Gulbenkian. (in portuguese)

Sárbu, I., & Borza, I. (1998). Energetic optimization of water pumping in distribution systems. *Mechanical Engineering*, *42*(2), 141–152.

Simpson, A. R., & Marchi, A. (2013). Evaluating the approximation of the affinity laws and improving the efficiency estimate for variable speed pumps. *Journal of Hydraulic Engineering*, *139*(12), 1314–1317.

Soares, J. S. M. (2015). *Numerical analysis and calibration methods for water supply systems* (Unpublished master's thesis). University of Aveiro.

Ulanicki, B., Kahler, J., & Coulbeck, B. (2008). Modeling the efficiency and power characteristics of a pump group. *Journal of Water Resources Planning and Management*, *134*, 88-93.

University of Exeter. (2008). *Operation benchmarks, centre for water systems resources.* `https://emps.exeter.ac.uk/engineering/research/cws/resources/benchmarks/`

`operation`.

van Zyl, J. E., Savic, D. A., & Walters, G. A. (2004). Operational optimization of water distribution systems using a hybrid genetic algorithm. *Journal of water resources planning and management*, *130*(2), 160–170.

Walski, T., Zimmerman, K., Dudinyak, M., & Dileepkumar, P. (2003). Some surprises in estimating the efficiency of variable-speed pumps with the pump affinity laws. In *World water and environmental resources congress.*

Walski, T. M., Chase, D. V., Savic, D. A., Grayman, W., Beckwith, S., & Koelle, E. (2003). *Advanced water distribution modeling and management*. Bentley Institute Press.

Part VI

Conclusion and closing remarks

16. General conclusions

The conclusions and final remarks related with each main topic addressed in this thesis are presented, as well as the general conclusions retrieved from the resulting product.

Water supply systems (WSS) are the target market addressed in this thesis and so, the main object of study. In such systems several challenges are identified such as (i) the need to reduce water losses and mostly, (ii) the need to reduce the energy consumption and improve the systems operational efficiency, which corresponds to 40 % of the water utilities concerns. The improvement of such systems is seen as a challenge due to the usually large dimensions of the networks and to the complexity of the operations that can be related with the high variability in the water demands and, sometimes, with the limited storage capacity. Such complexity of the operations can be also related with the large number and the diversity of components of the network that need to be controlled, such as several types of valves, pumps, storage tanks and sources. At the same time, any operational control modification results in changes in terms of pressure in the networks, which should always be supervised in order to maintain the minimum requirements for the consumers. Such complex tasks are not possible to be performed in the most efficient way by an operator. This will only be possible by transforming the conventional reactive control and monitoring into more pro-active control and monitoring. According to that, the WSS can present a huge opportunity for the implementation of emerging smart technologies.

According to the main objective proposed in this work, a numerical tool for the automatic efficiency improvement of any type of water supply and distribution network was developed. Such efficiency improvement may be performed by the tool using two complementary tasks: (i) the first one consists in the determination of the best controls of pumps and valves of the network in order to minimise the associated energy costs, which can be combined with a water demand prediction model for obtaining the near future best controls and (ii) the second one intends to improve the systems efficiency by reducing the energy needs through the implementation of hydroturbines for recovering excessive energy that is commonly dissipated in the networks. As shown in the results (Part V), this tool is capable of satisfactorily performing such tasks and demonstrated its capacity to be used in a large variety of water networks operating under distinct conditions.

Overall, this thesis presents particular research efforts in: (i) modelling and simulation of WSS, (ii) operational control optimisation of WSS, (iii) short-term water demand prediction and (iv) sites location and selection/design of hydroturbines for hydropower generation in WSS. Besides the detailed investigations performed for each addressed research topic, additional efforts were made for

the development of a product with utility and interest for the water industry.

16.1 Modelling and simulation of water supply systems

The hydraulic simulation module incorporated in the developed tool is essential for the reproduction of the WSS behaviour in order to test and assess the implementation of a certain efficiency measure. This module, based on the hydraulic simulator EPANET 2.0, performs some approximations that are analysed in this work.

Since the pumps play the most important role in terms of efficient control of the water networks, the methods used for modelling and simulating such elements are investigated. The new formulation proposed for the determination of the efficiency of pumps operating at variable speed demonstrates advantages in comparison with other formulations/methods such as (i) the Affinity Laws, (ii) the formulation proposed by Sárbu and Borza (1998) and (iii) the method used by EPANET that incorrectly applies the Affinity Laws. After analysing the use of such formulations/methods in the computation of savings from the application of efficiency measures in WSS, it is possible to conclude that the proposed formulation provides similar results with the Sárbu and Borza (1998) formulation. Concerning the accuracy in the prediction of the pumps best efficiency points, the proposed formulation presented the highest accuracy for a small pump. At the same time, in the prediction of an entire efficiency curve of a real pump, the proposed curve demonstrated also a superior accuracy.

The main drawback from using an hydraulic simulator in a tool as the one presented in this work is related with the computational time required for the simulation/evaluation of the networks behaviour, which may difficult a real-time implementation if the optimisation is required to be used in each time interval (1 hour, for instance). However, the developed tool is already capable of predicting optimal controls for each half a day to a day or more, if necessary. At the same time, the detailed simulation of a network is the only way to ensure a proper operation without violating the imposed requirements. The use of simplified models to predict the networks behaviour may significantly reduce the time of computation but, in turn, implies the implementation a new model and formulation for each different network, making this an impossible solution for the development of an automatic computational tool as the proposed in this thesis.

An additional advantage from the fact that the developed tool is incorporating the networks models is that several water utilities have already developed models of their networks. Some of them have in fact EPANET models and others may have different types of models that could eventually be imported to EPANET and thus, be capable of using the tool developed in this work.

16.2 Operational optimisation of water supply systems

The optimisation module, implemented for the optimal operational control of the water networks, incorporates a number of proposed techniques and innovative approaches.

The proposed problem formulation based on an explicit control, where the decision variables include (i) the pumps relative speeds and operating times and (ii) the valves opening times, demonstrated to be capable of obtaining superior results when compared with other works that tested the same benchmark network (van Zyl network). Such approach provided also proofs of its good performance in several distinct networks characterised by distinct operational conditions and distinct

designs and dimensions.

Overall, the PSO and DE algorithms demonstrated the best performance in the tested problems. However, the solutions found by the PSO algorithm are usually preferred to the ones obtained with the DE algorithm due to the significantly reduced number of switches. On the one hand, in the solutions obtained with the PSO algorithm, a tendency to results presenting only changes related with the pumps relative speeds was observed. On the other hand, the solutions obtained with the DE algorithm demonstrated both changes in terms of speed and operating/opening time. The consequence of many changes in the operating times using a probabilistic algorithm is the increase in the number of pumps and valves switches, leading to solutions that are not accepted by the water utilities. Besides the added difficulty in performing such type of daily control (since the pumps and valves have no automated control), the resulting excessive wearing of the equipment (both pumps and valves) would reduce significantly their life span and increase the need for maintenance, leading to a non-desirable increase in the long-term costs.

Compared to the nature-inspired metaheuristic algorithms, the Nelder-Mead Simplex algorithm presented the lowest performance, with less than 1 % of energy costs reduction in almost all tested benchmarks. However, it should be noticed that this algorithm is significantly faster than the nature-inspired and, in this work, the algorithm had the disadvantage of starting with a single solution while all the other algorithms (population-based) had a set of solutions instead.

The proposed approach for the aggregation of the decision variables, attempting to simplify the optimisation problem, revealed that, for some networks, the reduction of the universe of solutions caused by such aggregation technique may restrict the search to more complicated solutions that significantly increase the computational time required for the convergence of the hydraulic model. Thus, it was concluded that, even with a reduction in the number of variables and a reduction in the universe of solutions, the computational time for the entire process may significantly increase for some problems instead of decrease, as it was expected. This demonstrates the strong effect produced by the hydraulic simulation in the optimisation process.

For the tested problems, the sequential application of the algorithms did not demonstrate any advantage compared to the single algorithms.

16.3 Demand prediction in water supply systems

From the extensive literature review performed on the water demand forecasting topic, it is possible to conclude that the data analysis and pre-processing represents a very important role in the forecasting process with influence in the models' accuracy. This part of the process represents also the most time-consuming since a large amount of data is usually needed. At the same time, the collected data from the networks presents very often a large number of occurrences and/or missing data that, if not treated properly, can significantly influence the real data trends, reducing the accuracy of the models and, consequently, influencing any posterior attempt to optimise the operational control of the network.

It is also possible to conclude that, to develop an automatic forecasting model for operation in real-time, two main characteristics should be presented by the model: (i) scalability, to provide good performances for distinct model scales, and (ii) adaptability, to easily adapt to possible changes in the WSS conditions (self-learning capability) and avoid unnecessary model maintenance and calibration

(and thus, avoiding the associated costs).

In terms of the developed work related with water demand forecasting, although the implementation of forecasting models have already been performed, the scalability of the implemented methodologies, both for data processing and forecasting, was not tested since only data from one particular water supply system was used.

The traditional forecasting models (naïve and exponential smoothing) demonstrate variable performances for different data sets when predicting only one hour ahead. However, in the prediction of 24 hours ahead, the seasonal naïve forecasting models are more adequate for presenting lower performance errors. The models based on artificial neural networks are capable of improve such results if external input variables are introduced in the models. However, the influence that each variable (both anthropic and meteorological) is different for each data set and, most important, the wrong choice of the input variables may lead to a decrease in the forecasting model accuracy. It should then be concluded that a preliminary analysis of the input variables to include in a forecasting model is of the most importance.

16.4 Energy recovery in water supply systems

The process of determining if the installation of a hydropower scheme for energy recovery in a water supply system is viable (or even possible) is not easy since requires several efforts in terms of time and complexity of the feasibility studies, which may deviate the attention of the water utilities to this particularly interesting solution for the efficiency improvement of the water supply systems. For this reason, the inclusion of an option that, from the model of a network, automatically determines all the possibilities for the implementation of this type of solution while providing preliminary feasibility analysis could boost the adoption of this type of efficiency measures.

The developed module devoted to this particular task (energy recovery) makes use of the hydraulic model of the network to search by (and assess) locations for the installation of distinct types of turbines with the main objective of maximise the energy recovery. At the same time, technical and financial feasibility analyses results are provided, which includes the selection and preliminary design of appropriate turbines and the corresponding payback time of the project. The search for site locations is performed in all links of the network, which including the options of (i) install a turbine in the end of a certain pipe or (ii) replace a valve by a turbine.

The implemented methodology demonstrates to be effective through the comparison of results with other authors obtained for the model of a real network. Moreover, it was possible to conclude that, considering the maximisation of energy production (instead of maximisation of energy production and minimisation of water losses) as single objective, other sites in the network with higher potential can be located.

An additional conclusion from the obtained results is that the financial feasibility analysis can be decisive in the choice of a certain type of turbine for each particular location. A turbine of lower cost does not necessarily represent the fastest return of investment neither the highest revenues in a long-term period. In fact, such parameters are largely dependent on the site characteristics.

17. Contributions and recommendations

The main contributions of this work, both at scientific and industrial levels, are enumerated. Some recommendations and directions for future works are exposed.

The present thesis can be seen as a starting point for the development of a high potential software for the water sector. As it is, the developed tool may yet present some limitations, as already stated in Chapter 16. However, several significant contributions, both in scientific and industrial terms, were provided and, at the same time, the possibilities of extension of such tool are numerous.

17.1 The contribution of this thesis

This work is a contribution to Engineering that presents a combination of distinct disciplines such as fluid mechanics, hydraulic modelling, mathematical optimisation, control engineering, energy systems, data driven modelling, software development and engineering management.

In terms of scientific contributions, the main achievements of this thesis include: (i) an extensive literature review; (ii) a new formulation for the prediction of efficiency curves of variable-speed pumps; (iii) a new approach for the optimal operational control of water supply systems; (iv) a demonstration of the performance of different optimisation algorithms in several water supply and distribution networks; (v) a detailed analysis of the influence of several input variables in the accuracy of models for water demand forecasting; and (vi) a methodology for an automatic location of potential sites in the networks for energy recovery and preliminary selection/design of the most suitable turbine to be installed.

Concerning the industrial contributions, a factor of the most importance in engineering, the main achievements of the present work include: (i) a market assessment, demonstrating the adequacy of the developed work to the current and near future industry needs, (ii) a computer programme for the efficiency improvement in water supply systems; (iii) a computer programme that may present some common characteristics with the few ones existent in the world but applies state-of-the art techniques.

Moreover, the present thesis focuses on the connections between areas with high relevance for the efficiency improvement of water supply systems instead of dealing with each particular area as a separate research topic. Such connections provide the key to transform an academic work into a technology with potential to reach the industry.

As a result of the work developed for this thesis, the following papers were already published/accepted for publication in peer reviewed journals:

1. B. Coelho and A. Andrade-Campos (2016), A new approach for the prediction of speed-adjusted pump efficiency curves. *Journal of Hydraulic Research* (accepted for publication);
2. B. Coelho and A. Andrade-Campos (2016), Numerical tool for hydraulic modelling - An educational approach. *International Journal of Mechanical Engineering Education* (accepted for publication);
3. B. Coelho and A. Andrade-Campos (2014), Efficiency achievement in water supply systems - A review. *Renewable and Sustainable Energy Reviews*, 30, 59-84.

Part of the work was also presented in conferences and/or published in conference proceedings:

1. B. Coelho and A. Andrade-Campos (2014), On the comparison of numerical methodologies for control optimisation of variable-speed pumps. IWA World Water Congress and Exhibition, September 21-26th, 2014, Lisbon, Portugal (poster presentation);
2. B. Coelho and A. Andrade-Campos (2014). The operational costs minimisation in water supply systems using cascade optimisation techniques. IFORS 2014 - 20th Conference of the International Federation of Operational Research Societies, July, 2014, Barcelona, Spain (oral presentation);
3. B. Coelho and A. Andrade-Campos (2013), Improving water supply systems efficiency - A methodology for optimal control of variable-speed pumps, Proceedings of 2013 IAHR Congress, Tsinghua University Press, Beijing.

17.2 Recommendations and outlook

Starting with the topic that most influence any module of the tool developed in this work, the hydraulic modelling and simulation, a particular attention should be given to the performance of the hydraulic simulator used in the developed tool, the EPANET. Although EPANET is being used worldwide both by academics and professionals, there is a lack of studies comparing the performance of this simulator with others existent in the market. It would be interesting to test some of the most popular hydraulic simulators in order to find which one provides the most approximate representation of the real behaviour of the water supply systems. Such studies could also be useful, for instance, to improve possible limitations that can be revealed by EPANET.

The proper reproduction of the behaviour of variable-speed pumps becomes increasingly important with the increase of the use of such equipment, mostly due to the high impact this may reveal in the efficiency of the water supply systems. The main difficulty at this level is to find real data of the pumps (especially of the efficiency curves for distinct speeds of operation). Such data is usually not provided by the manufacturers neither by the water utilities. It would be interesting to observe a broader availability and sharing of such type of data.

Concerning the optimisation, a tendency of several researchers for applying Genetic Algorithms (GA) can be explained by the great performance of this algorithm (generally not the simple GA but an improvement of such algorithm) in several complex engineering problems. However, the investigation of other efficient and robust techniques remains open for future works.

The interest in the multi-objective optimisation applied to the control and design of water supply systems as also increased. However, this type of approach can be, in fact, approximated to the type of optimisation addressed in this work if the penalties added to the objective function are considered as additional objective functions. The main difference in a multi-objective approach is that a set of solutions (called pareto curve) is obtained, instead of a single-solution. However, this type of approach can also significantly increase the computational time. For some water utilities it may, in fact, be preferable to obtain different solutions (for example, (i) one that results in the minimum cost but very low pressures close to the limit and (ii) other that presents slightly higher costs but more desirable values of pressure, not so close to the limit). This can be obtained with a simple modification of the single-objective approach presented in this thesis by transforming the objective function into a weighted function with multiple objectives. Running multiple times such weighted function and changing the weights every time corresponds exactly to a multi-objective approach. In this case, to reduce the number of computations required, the use of three or four different weights may be enough.

An approach not tested in this work but that could be the best option to reduce the number of pump switches is to treat them as constraints. An additional constraint increases the complexity of the optimisation problem but, as already observed in the results presented in this work, such type of approach demonstrates to be effective even with the use of a simple penalty method. Additionally, the constraint handling techniques may be improved by using, for instance, dynamic penalties.

Despite the forecasting models developed in this work handled the different types of days as different input variables of the model, an interesting alternative approach could be the development of a different model for each type of day. Similarly with the work presented by Candelieri and Archetti (2014), each day can be associated to a different pre-defined type of day (week, weekend, holiday, ...) using, for instance, a clustering technique and then, the prediction of such day is performed according to the forecasting model trained for those type of days.

An additional feature that could be interesting to add to the energy recovery module of the developed tool is the inclusion of examples of real curves of turbines provided by manufactures and the automatic simulation of such types of turbines virtually placed in the selected sites in order to provide the feasibility analysis of viable solutions already existent in the market. Furthermore, the adaptation of the developed tool in order to efficiently control the entire network taking into account the energy produced by the turbines could also be a very interesting option.

For further steps at an industrial level, it would be important to improve the developed tool especially in the data processing and forecasting module, not forgetting to perform test with a large variety of data collected in different systems and with different measurement and communication equipment in order to prepare the tool for the largest possible range of eventualities. The implementation of an automatic calibration module connected to the hydraulic simulation module with scheduled updates of the network model is also a crucial step for the success of this type of tool. If the network behaviour is not modelled with an high level of accuracy, all the next steps for the real-time optimisation of the network may become useless and provide only advantages for the analysis of scenarios (which is still important but does not take full advantage of this type of tool).

It should not be forgotten that the adaptation of a data base (several water utilities already have one) to the numerical tool must be performed for each network, which, in turn, allows to respond to particular requirements of the water utilities.

Some pilot tests in real network(s) should be performed before the full implementation of this type of computer programme, considering only a selected section of the network capable of providing a larger flexibility for operational control changes while maintaining the required levels of security.

References

Candelieri, A., & Archetti, F. (2014). Identifying typical urban water demand patterns for a reliable short-term forecasting – The ICeWater project approach. *Procedia Engineering*, *89*, 1004–1012.

Sárbu, I., & Borza, I. (1998). Energetic optimization of water pumping in distribution systems. *Mechanical Engineering*, *42*(2), 141–152.

Appendices

A Design and operational optimisation methods overview

Authors	Year	Optimisation method	Constraints (in case of violation…)	Objectives	Type of control	Tested networks	Results (cost reduction, number of iterations, CPU time)	Source	Other observations
Cembrano et al.	1988	Conjugate Gradient	state and boundary constraints (penalty functions)	single-objective (min cost)	pump and valve control (24h - 1h intervals)	Barcelona (Spain)	1.7 $M, _, 24 min	Cembrano et al., 1988	no continuity
Brion & Mays	1991	Generalised Reduced Gradient (KYPIPE)	(penalties trhough Augmented Lagrangian method)	single-objective (min cost)	pump control (24h - 2h intervals)	Austin (Texas)	191 $ (-17.3%)	Brion & Mays, 1991	–
Mackle et al.	1995	GA	tank levels and continuity (penalty functions)	single-objective (min cost)	pump control (24h - 1h intervals) on/off	source - 4 pumps - reservoir	–	Mackle et al., 1995	obtained an optimised pump schedule adapted to the energy tariff variation
Savic & Walters	1997	GA	tank levels and continuity (progressive penalties technique)	multi-objective (min: energy costs; max: pump switches and feasibility)	pump control (24h -1h intervals) on/off	source - 4 pumps - reservoir	–	Savic & Walters, 1997	initial population based on previous runs improves the efficiency and the solutions
Goldman & Mays	1999	SA	tank levels, continuity, pressure bounds, water quality	single-objective (min cost)	pump control (24h-2h intervals) on/off	Austin / North Marin (California)	$221.38 (-4.1%) / $429.53	Goldman & Mays, 1999	the model can be easily adapted to a SCADA system
Atkinson et al.	2000	GA	–	–	–	Richmond (UK)	38300£ (-19%), 150000, 69 hours	Van Zyl et al., 2004	use of a commercial hydraulic simulator very time consuming which affected the final results
Cembrano et al.	2000	Generalised Reduced Gradient (WATERNET)	–	single-objective (min cost)	valve and pump control on/off	Sintra (Portugal)	(-18%)	Cembrano et al., 2000	Real-time operations (SCADA)
Pegg	2001	Decerto	–	single-objective (min cost)	half-hour control of fixed- variable- and dual-speed pumps	Wainuiomata-Waterloo network (Wellington)	(-10%)	Pegg, 2001	Real-time operations (Citect SCADA system)
Barnett et al.	2004	WaterGEMS	–	single-objective (min cost)	pump and valve control on/off	Jacksonville (Florida, USA)	–	Barnett et al., 2004	Real-time operations (SCADA)
Van Zyl et al.	2004	Pure GA	tank levels and number of pump switches	single-objective (min cost)	pump control on/off	case study / Richmond (UK)	350.36£, 100000 / 35296£ (-26%), 200000	Van Zyl et al., 2004	better performance of the hybrid using Hooke and Jeeves method than the Fibonacci
		Hybrid GA + Hill Climber (Fibonacci / Hooke&Jeeves) + EPANET					348.58£, 6000 / <355006£, 8000		
Carrijo et al.	2004	elitism-based SPEA (Strenght Pareto Evolutionary Algorithm) + EPANET2	–	multi-objective (min costs; max hydraulic benefits - index for pressure, demands and reservoir levels)	pump and valve control (24h - 1h intervals) on/off	Goiânia (Brazil)	good performance on finding the pareto-front	Carrijo et al., 2004	–
Lücken et al.	2004	Sequential and parallel implementation of MOGA, NPGA, NSGA, SPEA, NSGA-II and CNSGA-II	–	multi-objective (min: energy cost, number of pump switches, power peak and reservoir level variation)	Pump control on/off	source - pump station (5pumps) - reservoir	several improvements of parallel over sequential MOEAs	Lücken et al., 2004	–

Figure A.1: Operational optimisation (Part 1 of 2).

Authors	Year	Optimisation method	Constraints (in case of violation…)	Objectives	Type of control	Tested networks	Results (cost reduction, number of iterations, CPU time)	Source	Other observations
López-Ibáñez et al.	2005	SPEA2 + EPANET	tank levels and nodal pressures (constraint handling method based on ranking solutions)	multi-objective (min energy costs and number of pump switches)	pump control (24h - 1h intervals) on/off	Van Zyl	solutions that dominate the solution obtained by the correspondent algorithm in a single-optimisation approach	López-Ibáñez et al., 2005	–
Firmino et al.	2006	LP & ILP	continuity, max pump flow, reservoir levels	single-objective (min cost)	pump control on/off	Campina Grande (Brazil)	(-15%)	Firmino et al., 2006	–
Bunn	2007	Aquadapt	–	move energy use, reduce peaks and energy required by pumps	–	EBMUD / Washington	(-13.1%) / $1000	Bunn, 2007	Real-time operations
Salomons et al.	2007	DRAGA-ANN	pressure, tank levels, maximum pump power consumption and velocity	single-objective (min cost)	pump control (24h - 1h intervals) on/off	Haifa-A (Mount Carmel)	(-25%), _, faster than GA-EPANET	Salomons et al., 2007	Real-time operations (SCADA)
Martínez et al.	2007					Valencia (Spain)	(-17.6%)	Martínez et al., 2007	
López-Ibáñez et al.	2008	parallel ACO + EPANET	–	reduce CPU time	–	Richmond (UK)	_, 8000, <0.5h (around 2h with sequential ACO)	López-Ibáñez et al., 2008	in the parallel ACO, a higher number of ants reduces CPU time
Wang et al.	2009	GA	reservoir levels & flow in the system, to avoid underflow or overflow (penalty functions)	multi-objective (min cost, number of pump switches and total work time for each pump)	pump control on/off	source - pump station (several pumps) - reservoir	convergence speed could be improved	Wang et al., 2009	first work considering the land subsidence due to groundwater pumping all day long
Shihu et al.	2010	hybrid GSA (Genetic Simulated Annealing) + EPANET	tanks water levels, continuity, velocity limit of variable-speed pumps, number of switches	single-objective(min total cost = water production costs + electricity costs)	fixed- and variable-speed pumps control	City T (China)	(-6%)	Shihu et al., 2010	–
Bene & Hos	2011	MINLP, SBB / Series of Local Optima Technique, SLO	–	single-objective (min cost)	least-cost filling reservoir with a variable-speed pump without time constraints	source - pump - pipes - upper reservoir - consumption	SLO gives pump schedules with lower energy consumption (12.65 $) - better results for smaller time steps	Bene & Hos, 2011	decision variables - reservoir levels
López-Ibáñez et al.	2011	Simple Evolutionary Algorithm, SEA + EPANET	tank levels, continuity, pressure head, number of pump switches, EPANET warnings (constraint handling method based on ranking solutions)	single-objective (min cost)	pump control (24h - 1h intervals) on/off	Van Zyl /Richmond (UK)	315.9£ / _	López-Ibáñez et al., 2011	SEA with time-controlled triggers demonstrated better performance when compared to SEA with both binary and level-controlled triggers
Mouatasim et al.	2012	SPRG	population supply guarantee; tank levels	single-objective (min cost)	pump control on/off	Kénitra City /Agadir City (Morocco)	winter - 344.7$ / 937.44$ summer - 61.83$ / 278.37$	Mouatasim et al., 2012	–
		LINGO					winter - 516.3$ / 1001.25$ summer - 68.67$ / 278.37		

Figure A.2: Operational optimisation (Part 2 of 2).

New York City Tunnels							
Authors	Year	Method	Hydraulic simulation	Best cost ($ M)	Evaluations number	Feasible solution? (according to)	Sources	Other observations
Schaake et al.	1969	LP & DP	–	78.09	–	Yes (Dandy et al., 1996)	Dandy et al., 1996	–
Quindry et al.	1981	–	–	63.58	–		Geem, 2006	–
Gessler	1982	–	–	41.8	–	Yes (Dandy et al., 1996) No (Savic & Walters, 1997)	Dandy et al., 1996; Savic & Walters, 1997	–
Bhave	1985	–	–	40.18	–			–
Morgan & Goulter	1985	–	–	39.2	–	No (Dandy et al., 1996)	Dandy et al., 1996	–
Kessler	1988	–	–	39	–			–
Fujiwara & Khang	1990	–	–	36.1	–			–
Murphy et al.	1993	–	–	38.8	–	No (Savic & Walters, 1997)	Savic & Walters, 1997	–
Dandy et al.	1996	improved GA	KYPIPE	38.8	96750	No (Savic & Walters, 1997)	Dandy et al., 1996	–
Savic & Walters	1997	GA (GANET)	EPANET	37.13	1000000		Savic & Walters, 1997	w=10.5088
				40.42	–	Yes (Savic & Walters, 1997); No (Zecchin et al.,		w=10.9031
Lippai et al.	1999	Tabu Search	–	40.85	–	–	Geem, 2006	–
Wu & Simpson	2001	messy GA	EPANET	38.8	48387	–	Wu & Simpson, 2001	–
Cunha & Sousa	2001	SA	–	37.13	–	–	Geem, 2006	–
Wu & Simpson	2002	fast messy GA with self-adaptative boundary search	EPANET	38.8	30000	–	Wu & Simpson, 2010	–
Matias	2003	GA	–	38.64	–	–	Montalvo et al., 2008	–
Maier et al.	2003	ACO	Wadiso	38.64	13928	–	Maier et al., 2003	w=10.6668
Eusuff & Lansey	2003	SFLA	EPANET	38.13	–	–	Eusuff & Lansey, 2003	w=10.6668
Cunha & Ribeiro	2004	Tabu Search	–	37.1	–	No (Geem, 2009)	Cunha & Ribeiro, 2004	–
Geem	2006	HS	EPANET	36.66	6000	–	Geem, 2006	w=10.5088
Zecchin et al.	2007	AS(rank)	EPANET 2.0	38.64	19300	–	Zecchin et al., 2007	–
Montalvo et al.	2008	PSO	–	38.64	–	–	Montalvo et al., 2008	–
Perelman et al.	2008	Cross Entropy	–	38.64	–	–	Perelman et al., 2008	Multi-objective
Olsson et al.	2009	NSGA-II	–	38.64	–	–	Olsson et al., 2009	Multi-objective (results for zero deficits)
		UMDA	–	39.25	–	–		
		hBOA	–	44.56	–	–		
		CSM	–	46.40	–	–		
Geem	2009	PSHS	EPANET 1.0	38.64	4475	–	Geem, 2009	w=10.6668
vasan & Simonovic	2010	DE (DENET)	EPANET 2.0	38.64	30701	–	vasan & Simonovic, 2010	–
Bragalli et al.	2012	MINLP (Bonmin)	EPANET 2.0	36.38	–	–	Bragalli et al., 2012	w=10.5088

Figure A.3: Resume of the distinct methods, available in the literature, applied for the design optimisation of the New York City Tunnels network.

Hanoi								
Authors	Year	Method	Hydraulic simulation	Best cost ($ M)	Evaluations number	Feasible solution? (according to)	Sources	Other observations
Fujiwara & Khang	1990	NLPG & local improvement	_	5.562 6.320	_	No (Eiger et al., 1994)	Eiger et al., 1994 Geem, 2006	_
Eiger et al.	1994	Decomposition (Branch and Bound)	_	6.027	_	No (Savic & Walters, 1997)	Eiger et al., 1994	_
Savic & Walters	1995	GA	_	6.195	_	_	Montalvo et al., 2008	_
Savic & Walters	1997	GA (GANET)	EPANET	6.073	1000000	_	Savic & Walters, 1997 Geem, 2006	_
Abebe & Solomatine	1998	GLOBE	EPANET	7.006	16910	_	Abebe & Solomatine, 1998	_
Cunha & Sousa	1999	SA	_	6.056	53000	No (Eusuff & Lansey, 2003)	Geem, 2006	_
Wu et al	2001	GA	_	6.182	_	_	Montalvo et al., 2008	_
Matias	2003	GA	_	6.093	_	_		_
Cunha & Ribeiro	2004	Tabu Search	_	6.056	_	_	Cunha & Ribeiro, 2004	_
Liong & Atiquzzman	2004	SCE	EPANET	6.220	25402	_	Liong & Atiquzzman, 2004	_
Geem	2006	HS	EPANET	6.056	200000	_	Geem, 2006	w=10.6668
Zecchin et al.	2007	MMAS	EPANET 2.0	6.134	85600	_	Zecchin et al., 2007	_
Montalvo et al.	2008	PSO	_	6.133	_	_	Montalvo et al., 2008	_
Geem	2009	PSHS	EPANET 1.0	6.081	17980	_	Geem, 2009	w=10.6668
Vasan & Simonovic	2010	DE (DENET)	EPANET 2.0	6.056	50201	_	Vasan & Simonovic, 2010	_
Bragalli et al.	2012	MINLP (Bonmin)	EPANET 2.0	6.056	_	_	Bragalli et al., 2012	_

Figure A.4: Hanoi water network

Two-reservoir								
Authors	Year	Method	Hydraulic simulation	Best cost ($ M)	Evaluations number	Feasible solution? (according to)	Sources	Other observations
Gessler	1985	Selective Enumeration	_	1833	_	_	Simpson et al., 1994	_
Simpson et al.	1994	Complete enumeration	_	1750	11940	_		_
		NLO (GINO)	Wadiso	1760	_	_		_
		GA	Newton-Raphson solver	1750	50000	_		_
Wu & Simpson	2001	messy GA	_	1750	2400	_	Wu & Simpson, 2001	_
Maier et al.	2003	ACO	Wadiso	1750	8509	_	Maier et al., 2003	_
Cheung et al.	2003	SPEA	EPANET 2.0	1667	_	_	Cheung et al., 2003	2 objectives: cost & pressure deficit minimisation
Zecchin et al.	2007	AS(elite)	EPANET 2.0	1750	1800	_	Zecchin et al., 2007	_
		AS(rank)		1750	1500	_		_

Figure A.5: Two reservoir water network

Anytown								
Authors	Year	Method	Hydraulic simulation	Best cost ($ M)	Evaluations number	Feasible solution? (according to)	Sources	Other observations
Gessler	1985	Enumeration	_	12.3	_	_	Walski et al., 1987	_
Lee et al.		Gradient search & LP	_	12.9	_	_		_
Morgan & Goulter		Hardy-Cross method & LP	_	13	_	_		_
Ormsbee		Box-Complex search	_	13.8	_	_		_
Murphy et al.	1994	_	_	11.4	_	_		_
Walters et al.	1999	SMGA (2 objectives: max benefit & min cost)	_	10.9 (cheapest)	_	_	Walters et al., 1999	pumping and storage included
			_	11 (preferred)	_	_		
Farmani et al.	2006	Evolutionary Multi-objective (max reliability, min cost & residence time)	EPANET 2.0	13.4 (cheapest)	_	_	Farmani et al., 2006	pump operation schedules included
Olsson et al.	2009	NSGA-II	_	20.6	_	_	Olsson et al., 2009	Multi-objective (results for zero deficits)
		UMDA		15.9				
		hBOA		17.9				
		CSM		16.1				

Figure A.6: Anytown water networks

Two-loop								
Authors	Year	Method	Hydraulic simulation	Best cost ($ M)	Evaluations number	Feasible solution? (according to)	Sources	Other observations
Alperovits & Shamir	1977	Decomposition	_	0.498	_	_	Alperovitz & Shamir, 1977	_
Gouler et al.	1986	Decomposition	_	0.435	_	_	Geem, 2009	_
Kessler & Shamir	1989	Decomposition (matrix notation)	_	0.418	_	_	Kessler & Shamir, 1989	_
Fujiwara & Khang	1990	_	_	0.415	_	_	Djebedjian et al., 2000	_
Eiger et al.	1994	Decomposition (Branch and Bound)	_	0.402	_	-	Eiger et al., 1994	_
Savic & Walters	1997	GA (GANET)	EPANET	0.419	25000	_	Savic & Walters, 1997	_
Abebe & Solomatine	1998	GLOBE	EPANET	0.419	1373	_	Abebe & Solomatine, 1998	_
Cunha & Sousa	1999	SA	_	0.419	70000	_	Djebedjian et al., 2000 Geem, 2009	w=10,5088
Djebedjian et al.	2000	SUMT	_	0.419	_	_	Djebedjian et al., 2000	_
Wu et al.	2001	GA	_	0.419	7467	_	Geem, 2009	w=10,5088
Eusuff & Lansey	2003	SFLA	EPANET	0.419	11155	_	Eusuff & Lansey	w=10,6668
Cunha & Ribeiro	2004	Tabu Search	_	0.42	_	_	Cunha & Ribeiro, 2004	_
Liong & Atiquzzaman	2004	SCE	EPANET	0.419	1091	_	Liong & Atiquzzaman, 2004	_
Geem	2006	HS	EPANET	0.419	1067	_	Geem, 2006	w=10,6668
Geem	2009	PSHS	EPANET 1.0	0.419	204	_	Geem, 2009	w=10,6668
Formiga et al.	2006	NSGA-II	Gradient method	0.450	5937	_	Formiga et al., 2006	use of D-W formula instead of H-W and leakages consideration

Figure A.7: Two loop water network

B Short-term water demand forecasting methods overview

Authors, year	Time scale [amount of	Model scale [population	Forecasting model	training R^2 (-)	testing R^2 (-)	RMSE (m^3/h)	MAE (m^3/h)	MARE / MAPE (%)	maxARE (%)	NSE (-)	Observations
Jain et al., 2001	Weekly [98weeks]	indian institute [12k]	MLR	0.684	0.642	_	_	5.16	10.39	_	_
			MNLR	0.667	0.602	_	_	5.59	10.52	_	
			AR	0.826	0.624	_	_	17.68	24.07	_	
			ANN (1HL) – FFBP	0.963	0.640	_	_	3.74	12.08	_	
			ANN (2HL) - FFBP	0.992	0.872	_	_	2.41	6.49	_	
Bougadis et al., 2005	Peak weekly [4months]	City of Otawa, Canada [0.75M]	MLR	0.620	0.445	_	_	19.25	43.18	_	_
			ARIMA(2,1,0)	0.300	0.352	_	_	14.31	36.28	_	
			ANN (1HL) – FFBP - sigmoid	0.708	0.810	_	_	12.26	30.05	_	
Alvisi et al., 2007 [POWADIMA]	Hourly [1year]	Castelfranco Emilia, Italy [23k]	Pattern-based model with periodic and persistence components	_	_	16.2-28.8	_	5.0-9.0	_	_	The distinct case-studies show the scale effect in the models performance when comparing through a scale dependent meaure
Salomons et al., 2007 [POWADIMA]	Hourly [1year]	Haifa-A, Israel [60k]	Pattern-based model with periodic and persistence components (1h/24h lead time)	_	_	131.0-155.5	_	8.6 / 10.3	_	_	
Martinez et al., 2007 [POWADIMA]		Valencia, Spain [1.2M]		_	_	784.8-860.4	_	4.7 / 5.1	_	_	
Adamowski, 2008	Peak daily summer [10years -	City of Ottawa, Canada [0.75M]	MLR	0.610	0.590	_	_	14.00	56.00	_	_
			ARIMA(2,1,0)	0.530	0.460	_	_	15.00	62.00	_	
			ANN (1HL) - FFBP	0.660	0.690	_	_	12.00	41.00	_	
Ghiassi et al., 2008	Hourly [1month - Sep/Apr]	City of San Jose & surroundings, California [0.9M]	ARIMA	_	_	_	_	3.38 / 4.51	_	_	_
			ANN - FFBP	_	_	_	_	3.41 / 4.52	_	_	
			dynamic ANN	_	_	_	_	2.04 / 3.26	_	_	
	Daily [1year - 366days]		ARIMA	_	_	_	_	2.35	_	_	
			ANN - FFBP	_	_	_	_	2.32	_	_	
			dynamic ANN	_	_	_	_	0.84	_	_	
	Weekly [4years - 208weeks]		ARIMA	_	_	_	_	1.17	_	_	
			ANN - FFBP	_	_	_	_	0.92	_	_	
			dynamic ANN	_	_	_	_	0.80	_	_	
Msiza et al., 2008	Daily [9years & 5months -	Gauteng Province, South Africa [9M]	ANN - RBF	_	_	_	_	2.96	_	_	_
			SVM	_	_	_	_	5.47	_	_	
Tabesh & Dini, 2009	Daily [13years]	Tehran, Iran [8/12M, night/day]	ANN (3HL)	_	0.943	0.28	_	2.74	_	_	_
			ANN-RBF	_	0.932	0.30	_	2.94	_	_	
			Fuzzy	_	0.760	0.74	_	7.62	_	_	
			Neural-Fuzzy (non-random input data)	_	0.801	0.32	_	2.77	_	_	
			Neural-Fuzzy (random input data)	_	0.936	0.30	_	2.86	_	_	
Adamowski & Karapataki, 2010	Peak weekly [6years]	Athalassa, Nicosia, Cyprus [0.2M in the city]	MLR	0.838	0.820	8.13	_	2.51	11.99	_	_
			ANN(1HL, 15HN) - Resilient BP	0.940	0.901	6.99	_	2.23	12.15	_	
			ANN(1HL, 15HN) - CGPB	0.947	0.942	6.77	_	2.09	11.93	_	
			ANN(1HL, 15HN) - LM	0.953	0.946	5.23	_	2.15	11.18	_	
		Public Garden, Nicosia, Cyprus [0.2M in the city]	MLR	0.839	0.813	8.05	_	2.48	12.59	_	_
			ANN(1HL, 15HN) - Resilient BP	0.942	0.900	6.90	_	2.27	10.41	_	
			ANN(1HL, 15HN) - CGPB	0.935	0.905	7.26	_	2.21	11.26	_	
			ANN(1HL, 15HN) - LM	0.957	0.917	5.69	_	1.97	9.76	_	
Herrera et al., 2010	Hourly [4months]	WSS in a city, Spain [5k]	ANN(1HL) - FFBP (grow/slide data update)	_	_	_	6.23 / 6.30	_	_	_	No significant difference when updating data by accumulating (grow) or considering only last observations (slide). Paper do not provide RMSE nor MAE units*.
			Project Pursuite Regression	_	_	_	4.36 / 4.36	_	_	_	
			Multivariate Adaptive Regression Splines	_	_	_	4.36 / 4.36	_	_	_	
			Random Forests (regression trees)	_	_	_	4.36 / 4.36	_	_	_	
			Pattern-based	_	_	_	9.32 / 9.61	_	_	_	
			SVR	_	_	_	4.32 / 4.32	_	_	_	
			SVR (best model with other data)	_	_	4.38	3.24	_	_	0.445	
Babel & Shinde, 2011	Daily [1066days]	Bangkok, Thailand [7.91M]	ANN (3HL) - FF - Momentum (gradient-descent) – tanh	_	_	2083.32	_	1.12	_	_	Meteorological variables have more influence on monthly forecasts
Odan & Reis, 2012	Hourly [10months]	Araraquara city, Brazil [224k]	ANN (MLP) - BP	0.846	0.740	_	16.20	_	_	_	_
			dynamic ANN	0.757	0.810	_	12.60	_	_	_	
			Hybrid ANN (MLP) - BP (Fourier Series as input)	0.792	0.740	_	16.20	_	_	_	
			Hybrid dynamic ANN (FS input)	0.846	0.865	_	10.44	_	_	_	
Adamowski et al., 2012	Daily summer [8years & 3months - May to Aug]	City of Montreal, Canada [1.8M]	MLR	0.76	0.786	_	_	_	_	0.629	_
			MNLR	0.848	0.838	_	_	_	_	0.633	
			ARIMA	0.758	0.782	_	_	_	_	0.778	
			ANN	0.792	0.865	_	_	_	_	0.864	
			WANN (wavelets as input)	0.896	0.919	_	_	_	_	0.919	
Tiwari & Adamowski, 2013	Daily [4179days]	City of Montreal, Canada [1.8M]	ARIMA	0.900	_	271.26	202.07	_	_	_	_
			ARIMA with exogenous input (maxT and totP)	0.910	_	272.92	196.67	_	_	_	
			ANN (13HN)	0.920	_	245.02	173.74	_	_	_	
			BANN (bootstrap)	0.900	_	253.76	183.74	_	_	_	
			WANN	0.980	_	129.56	92.09	_	_	_	
			WBANN	0.980	_	170.42	124.60	_	_	_	
	Weekly [138months]		ARIMA	0.680	_	489.17	312.08	_	_	_	_
			ARIMA with exogenous input (maxT and totP)	0.680	_	495.43	366.66	_	_	_	
			ANN (5HN)	0.670	_	492.91	357.91	_	_	_	
			BANN	0.680	_	513.76	382.07	_	_	_	
			WANN	0.760	_	502.92	370.84	_	_	_	
			WBANN	0.780	_	419.18	312.08	_	_	_	

Figure B.1: Comparison of the performances of short-term water demand forecasting methods applied to distinct case-studies (Part 1 of 2).

Authors, year	Time scale [amount of	Model scale [population	Forecasting model	training R^2 (-)	testing R^2 (-)	RMSE (m3/h)	MAE (m^3/h)	MARE / MAPE (%)	maxARE (%)	NSE (-)	Observations
Adamowski et al., 2014; Tiwari & Adamowski, 2014	Weekly [2years & 9months]	City of Calgary, Canada [1.1M]	ANN (6HN)	0.590	_	2502.07	1849.57	_	_	_	_
			BANN (6HN) - bootstrap of 100ANN	0.560	_	2442.92	1772.10	_	_	_	
			WANN (4HN)	0.730	_	1899.58	1424.16	_	_	_	
			WBANN (4HN) - bootstrap of 100WANN	0.800	_	1666.66	1218.35	_	_	_	
Santos & Filho, 2014	Hourly (1 to 24h forecast) [1year]	Cantareira WSS, São Paulo, Brazil [6.5M**]	MLR	0.442	_	4853	3715.20	_	_	_	paper do not provide RMSE nor MAE units*. Best model obtained for predicting 12h ahead using anthropic, weather and past
			ANN(1HL) - FFBP	0.481 - 0.679	0.454 - 0.500	7380 - 3722	5634-2761	_	_	_	
Romano & Kapelan, 2014	Hourly (24h ahead), [6months]	DMA1 of Yorkshire WSS, UK	EA-ANN (with / without data update)	_	_	86.08 / 207.94	_	6.39 / 8.47	_	0.96 / 0.94	_
			fixed-structure ANN	_	_	561.60 / 786.41	_	9.83 / 11.93	_	0.91 / 0.89	
			ensemble EA-ANN	_	_	82.94 / 128.30	_	6.30 / 7.73	_	0.96 / 0.96	
			ensemble fixed-structure ANN	_	_	370.15 / 370.88	_	9.07 /10.54	_	0.93 / 0.93	
		DMA3 (less consumers) of Yorkshire WSS, UK	EA-ANN (with/without data update)	_	_	0.13 / 0.14	_	5.64 / 5.51	_	0.98 / 0.97	
			fixed-structure ANN	_	_	1.05 / 0.86	_	9.24 / 9.72	_	0.93 /0.94	
			ensemble EA-ANN	_	_	0.13 / 0.14	_	5.68 / 5.51	_	0.98 / 0.98	
			ensemble fixed-structure ANN	_	_	0.76 / 0.86	_	8.68 / 9.72	_	0.94 / 0.94	
		reservoir outlet of Yorkshire WSS, UK [70k]	EA-ANN (with/without data update)	_	_	_	_	6.23 / 7.36	_	0.94 / 0.93	
			fixed-structure ANN	_	_	_	_	8.84 / 9.37	_	0.91 / 0.91	
			ensemble EA-ANN	_	_	_	_	5.68 / 6.26	_	0.95 / 0.95	
			ensemble fixed-structure ANN	_	_	_	_	7.75 / 8.63	_	0.93 / 0.92	
Bakker et al., 2013 & Bakker, 2014	48h forecast with 15-min time steps [6years - 210336values]	Amsterdam, Netherlands	adaptive pattern-based, 24-h	_	_	151.55	_	1.44	_	0.785	better accuracy for larger areas / higher NSE (better fit) when using 15-min time steps
			adaptive pattern-based, 15-min	_	_	365.69	_	3.35	_	0.987	
		Rijnregio, Netherlands	adaptive pattern-based, 24-h	_	_	63.80	_	1.86	_	0.710	
			adaptive pattern-based, 15-min	_	_	165.70	_	4.64	_	0.978	
		Almere, Netherlands	adaptive pattern-based, 24-h	_	_	36.19	_	2.12	_	0.740	
			adaptive pattern-based, 15-min	_	_	100.69	_	5.28	_	0.972	
		Helden, Netherlands [39k]	adaptive pattern-based, 24-h	_	_	15.04	_	3.4	_	0.803	
			adaptive pattern-based, 15-min	_	_	30.03	_	6.55	_	0.952	
		Valkenburg, Netherlands [9.2k]	adaptive pattern-based, 24-h	_	_	3.53	_	3.49	_	0.802	
			adaptive pattern-based, 15-min	_	_	7.30	_	6.90	_	0.949	
		Hulsberg, Netherlands [2.4k]	adaptive pattern-based, 24-h	_	_	1.48	_	5.12	_	0.658	
			adaptive pattern-based, 15-min	_	_	3.01	_	10.44	_	0.905	
Bakker et al., 2014 & Bakker, 2014	Daily [6years - 2192days]	Amsterdam (urban), Netherlands	MLR (without/with weather input)	_	_	_	_	1.54/1.47	_	0.709 / 0.738	rural areas more difficult to predict / weather variables input improve the models performance in all cases
			adaptive pattern-based, 1day	_	_	_	_	1.39/1.32	_	0.756 / 0.790	
			transfer-/noise (transfer model with ARIMA(0,8,3))	_	_	_	_	1.36/1.25	_	0.766 / 0.812	
		Rijnregio (mix), Netherlands	MLR	_	_	_	_	1.99/1.87	_	0.681 / 0.731	
			adaptive pattern-based, 1day	_	_	_	_	1.88/1.73	_	0.694 / 0.751	
			transfer-/noise	_	_	_	_	1.77/1.65	_	0.740 / 0.777	
		Almere (urban), Netherlands	MLR	_	_	_	_	2.37/2.26	_	0.681 / 0.731	
			adaptive pattern-based, 1day	_	_	_	_	2.08/1.97	_	0.718 / 0.764	
			transfer-/noise	_	_	_	_	2.03/1.87	_	0.733 / 0.793	
		Helden (rural), Netherlands	MLR	_	_	_	_	4.25/4.01	_	0.747 / 0.780	
			adaptive pattern-based, 1day	_	_	_	_	3.72/3.33	_	0.794 / 0.848	
			transfer-/noise	_	_	_	_	3.74/3.47	_	0.791 / 0.832	
		Valkenburg (rural), Netherlands	MLR	_	_	_	_	3.97/3.81	_	0.756 / 0.772	
			adaptive pattern-based, 1day	_	_	_	_	3.55/3.38	_	0.788 / 0.808	
			transfer-/noise	_	_	_	_	3.44/3.32	_	0.807 / 0.818	
		Hulsberg (rural), Netherlands	MLR	_	_	_	_	5.97/5.73	_	0.619 / 0.643	
			adaptive pattern-based, 1day	_	_	_	_	5.08/4.48	_	0.688 / 0.755	
			transfer-/noise	_	_	_	_	5.04/4.77	_	0.696 / 0.727	
Wang et al., 2014	Hourly [1year]	Barcelona, Spain [3M]	Double-Season multiplicative Holt-Winters + Gaussian Process regression	_	_	3.99-5.47	3.06-4.32	_	_	_	_
Candelieri & Archetti, 2014	Hourly [13months]	Milan, Italy [1M]	data clustering + SVM	_	_	_	_	0.79-14.33	_	_	_
Kang et al., 2015	Hourly [6months]	WSS in Gallella (rural), Sri Lanka	ARIMA(1,1,2)	_	0.83	6.21	_	_	_	0.80	_
			ARIMA(1,1,2) + Exponential Smoothing	_	0.90	5.26	_	_	_	0.88	

MLR - Multiple Linear Regression
MNLR - Multiple Non-Linear Regression
AR - Autoregression
ANN - Artificial Neural Networks
FF - Feed Forward
BP - Back Propagation
HL - Hidden Layer
SVM - Support Vector Machine
RBF - Radial Basis Function
CG - Conjugate Gradient
LM - Levenberg-Marquardt
SVR - Support Vector Regression
WBANN - Wavelet Bootstrap ANN
ARIMA - Autoregressive Integrated Moving Average

* in order to convert for l/s units, it was considered that the authors used the same units of water consumption.
** number of consumers supplied obtained from *Complexo Metropolitano - sabesp* (2015) website.

Figure B.2: Comparison of the performances of short-term water demand forecasting methods applied to distinct case-studies (Part 2 of 2).

Authors & year	Time scale	Model scale	Tested input variables	Best forecasting model / other observations
Jain et al., 2001	Weekly	indian institute	WD(t-2, t-1), maxT(t-1, t), R(t-1, t), RO(t-1, t)	ANN(1HL) / ANN(2HL): WD(t-1), maxT(t), RO(t) AR: WD(t-2, t-1) MLR: WD(t-1), maxT(t), RO(t) MNLR: WD(t-1), maxT(t-1, t), R(t-1, t)
Bougadis et al., 2005	Peak weekly	City of Otawa, Canada	WD(t-3 to t), maxT(t-1, t), WD(t-3 to t), maxT(t-1, t), R(t-1, t), RO(t-1, t)	ANN(1HL): WD(t-1), maxT(t), R(t) ARIMA(2,1,0): WD(t-3 to t-1) MLR: WD(t-1), maxT(t-1,t), R(t-1,t)
Adamowski, 2008	Peak daily	City of Otawa, Canada	WD(t-3 to t), maxT(t-1, t), R(t-5 to t), RO(t-5 to t)	ANN(1HL) / MLR: WD(t-1), maxT(t-1,t), RO(t-5) ARIMA(2,1,0): WD(t-3 to t-1)
Msiza et al., 2008	Daily	Gauteng province, South Africa	WD(t-5 to t), annual Pop.	ANN: WD(t-3 to t) and annual Pop.
Tabesh & Dini, 2009	Daily	Tehran, Iran	WD(t-7 to t), previous week and previous year total WD, avgT, RH	ANN/Neuro-fuzzy: WD(t-7 to t), previous week and previous year total WD
Adamowski & Karapataki, 2010	Peak weekly	Athalassa, Nicosia Public Garden, Nicosia	WD, maxT, R, RO	MLR: WD(t-1), maxT(t-2 to t) ANN-LM(1HL, 15HN): WD(t-1), maxT(t-1, t), R(t-1, t), RO(t-1, t) MLR: WD(t-1), maxT(t-1, t), R(t-1, t), RO(t-1, t) ANN-LM(1HL, 15HN): WD(t-1), maxT(t-2 to t)
Herrera et al., 2010	Hourly	Spain WSS	WD(t-168+1, t-1, t), R, T, windS, Press	–
Babel & Shinde, 2011	Daily	Bangkok, Thailand	WD(t-6 to t), R(t), Evap(t), RH(t), maxT(t), minT(t), avgT(t)	WD(t), R(t), avgT(t), RH(t)
Odan & Reis, 2012	Hourly	1WSS subsector, Sao Paulo, Brazil	WD(t-168, t-24, t-3 to t), T(t), RH(t) and FS	ANN(MLP)-BP (8HN): WD(t-168, t-3 to t), RH(t) dynamic ANN (15HN): WD(t-168, t-2 to t) hybrid ANN (8HN): WD(t-168, t-3 to t), FS(t-168, t-3 to t), RH(t) hybrid dynamic ANN (15HN): WD(t-168, t-2 to t), FS(t-168, t-2 to t)
Adamowski et al., 2012	Daily	City of Montreal, Canada	maxT, totP & WD (t-3 to t)	MLR: WD(t-1, t) & maxT(t-1, t) MNLR: WD(t-3 to t) & maxT(t-3 to t) ANN: WD(t-2 to t) & maxT(t-1, t) WANN: WD(t-3 to t) & maxT(t-1, t)
Tiwari & Adamowski, 2013	Daily Weekly	City of Montreal, Canada	WD(t-6 to t), maxT(t-6 to t), totP(t-6 to t) + 4 wavelet components of each	WANN: all 4 wavelet components of WD(t) WBANN: all 4 wavelet components of WD(t), 2 wavelet components of maxT(t-3 to t-1) and of totP(t-3 to t-1)
Adamowski et al., 2014	Weekly	Calgary city, Canada	WD(t-3 to t), maxT(t-3 to t), totP(t-3 to t) + 4 wavelet components for each series	WBANN:all 4 wavelet components of WD(t), 2 wavelet components of maxT(t-3 to t-1) and 1 wavelet component of totP(t-3 to t-1)
Santos & Filho, 2014	Hourly	39 cities in São Paulo, Brazil	demand(t-1, t-6, t-12, t-18, t-24); WD / anthropic(t, t+6, t+12, t+18, t+24); Hour, Day, Seas, typeDay / weather(t, t-1, t-6, t-12, t-18, t-24); T, RH, R, Press, windDir, windS	Output: WD(t+12) / Input: anthropic(t+12), weather(t, t-12), WD(t, t-12) [best model] Output: WD(t) / Input: anthropic(t), weather(t, t-1), WD(t-1) [worst than MLR]

Figure B.3: Comparison of the influence of several variables as input for distinct short-term water demand forecasting models

C Roughness coefficients for the computation of headlosses in pipes

Table C.1: Hazzen-Williams C-factors (or roughness coefficients) for various pipe materials (adapted from Walski et al., 2001). Note that a $2.5 - 122$ cm diameter range is covered in the presented values (values are larger for large pipe diameters).

Pipe material	C-factor [-]
Uncoated cast iron - smooth and new	121 - 134
Coated cast iron - smooth and new	129 - 141
Coated cast iron - 30 years old	41 - 120
Coated cast iron - 60 years old	30 - 112
Coated cast iron - 100 years old	21 - 104
Miscellaneous - newly scraped/brushed	97 - 127
Galvanised iron - smooth and new	120 - 133
Wrought iron - smooth and new	129 - 142

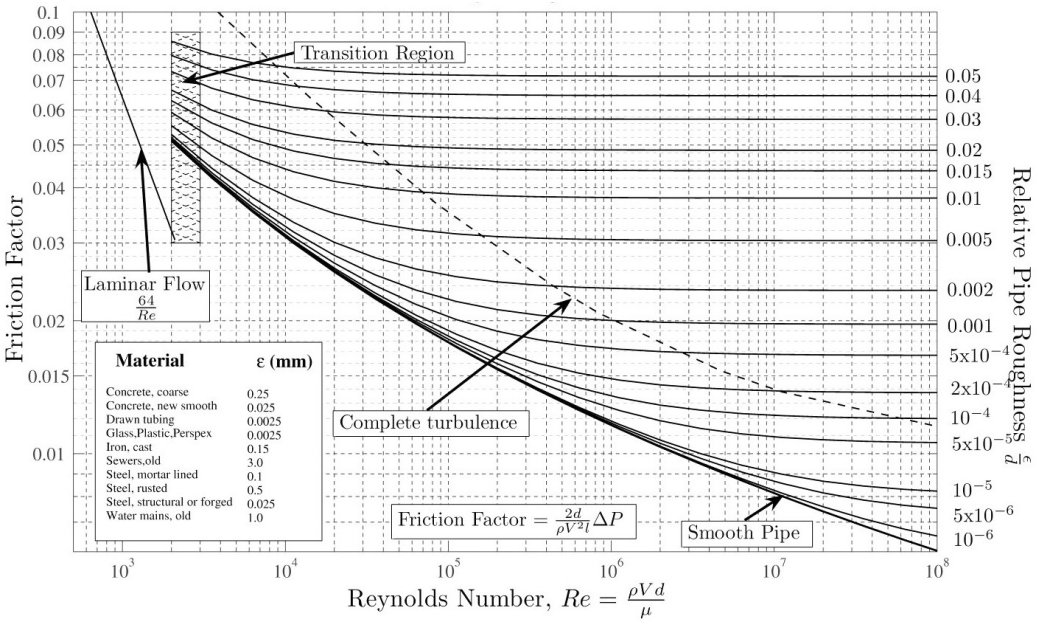


Figure C.1: Moody Diagram showing the Darcy-Weisback friction factor for different Reynolds numbers, *Re*, and different relative D-W pipe roughness, $\frac{\varepsilon}{d}$ (White, 2011).

D EPANET input files

D.1 Single-pump network - EPANET input file

```
[TITLE]
Single-pump network | bcoelho | 2015

[TITLE]


[JUNCTIONS]
;ID              Elev         Demand       Pattern
 node            10           0                       ;
 1               90           10           2          ;

[RESERVOIRS]
;ID              Head         Pattern
 source          10                        ;

[TANKS]
;ID              Elevation    InitLevel    MinLevel     MaxLevel     Diameter     MinVol       VolCurve
 tank            100          4            1            10           7            0                         ;

[PIPES]
;ID              Node1            Node2           Length       Diameter     Roughness    MinorLoss    Status
 pipe            node             tank            2000         200          50           0            Open   ;
 1               tank             1               2000         200          50           0            Open   ;

[PUMPS]
;ID              Node1            Node2           Parameters
 pump            source           node            HEAD 1 PATTERN 1 ;

[VALVES]
;ID              Node1            Node2           Diameter     Type Setting      MinorLoss

[TAGS]

[DEMANDS]
;Junction        Demand       Pattern          Category

[STATUS]
;ID              Status/Setting

[PATTERNS]
;ID              Multipliers
;pump pattern
 1             1.0000          1.0000          1.0000          1.0000          1.0000        1.0000
 1             1.0000          1.0000          1.0000          1.0000          1.0000        1.0000
 1             1.0000          1.0000          1.0000          1.0000          1.0000        1.0000
 1             1.0000          1.0000          1.0000          1.0000          1.0000 1.0000
;demand pattern
 2             2.0000          2.0000          2.0000          2.0000          2.0000            2.0000
 2             4.0000          4.0000          2.0000          2.0000          2.0000            2.0000
 2              3.0000          3.0000          2.0000          2.0000          2.0000            2.0000
 2             3.0000          3.0000          3.0000          4.0000          2.0000            2.0000
;Energy Tariff
 3               0.0700          0.0700          0.0700          0.0700         0.0700         0.0700
 3               0.0700          0.1000          0.1000          0.1000          0.1000          0.1000
 3               0.1000          0.1000          0.1000          0.1000          0.1000  0.1000
 3               0.1000          0.1000          0.1000          0.1000          0.0700          0.0700

[CURVES]
;ID              X-Value      Y-Value
;PUMP: PUMP: PUMP: PUMP: PUMP: Pump curve
 1               0            200
 1               50           150
 1               100          0
;EFFICIENCY: pump efficiency curve
 2               0            0
 2               35           60
 2               50           75
 2               60           78
 2               75           77
 2               80           74
 2               85           70
 2               90           65
 2               95           60
 2               150          0

[CONTROLS]
LINK pump 0.000 AT TIME 0.0000
LINK pump 1.000 AT TIME 1.0000
```

```
LINK pump 1.000 AT TIME 2.0000
LINK pump 1.000 AT TIME 3.0000
LINK pump 1.000 AT TIME 4.0000
LINK pump 1.000 AT TIME 5.0000
LINK pump 1.000 AT TIME 6.0000
LINK pump 0.000 AT TIME 7.0000
LINK pump 0.000 AT TIME 8.0000
LINK pump 0.000 AT TIME 9.0000
LINK pump 0.000 AT TIME 10.0000
LINK pump 0.000 AT TIME 11.0000
LINK pump 0.000 AT TIME 12.0000
LINK pump 0.000 AT TIME 13.0000
LINK pump 0.000 AT TIME 14.0000
LINK pump 0.000 AT TIME 15.0000
LINK pump 0.000 AT TIME 16.0000
LINK pump 0.000 AT TIME 17.0000
LINK pump 0.000 AT TIME 18.0000
LINK pump 1.000 AT TIME 19.0000
LINK pump 1.000 AT TIME 20.0000
LINK pump 1.000 AT TIME 21.0000
LINK pump 1.000 AT TIME 22.0000
LINK pump 1.000 AT TIME 23.0000

[RULES]

[ENERGY]
 Global Efficiency   75
 Global Price        1
 Global Pattern      3
 Demand Charge       0
 Pump  pump            Efficiency 2
 Pump  pump            Price      1
 Pump  pump            Pattern    3

[EMITTERS]
;Junction         Coefficient

[QUALITY]
;Node             InitQual

[SOURCES]
;Node             Type         Quality      Pattern

[REACTIONS]
;Type      Pipe/Tank        Coefficient

[REACTIONS]
 Order Bulk             1
 Order Tank             1
 Order Wall             1
 Global Bulk            0
 Global Wall            0
 Limiting Potential     0
 Roughness Correlation  0

[MIXING]
;Tank             Model

[TIMES]
 Duration            24:00
 Hydraulic Timestep  1:00
 Quality Timestep    0:05
 Pattern Timestep    1:00
 Pattern Start       0:00
 Report Timestep     1:00
 Report Start        0:00
 Start ClockTime     0 am
 Statistic           NONE

[REPORT]
 Status              Full
 Summary             No
 Page                0
 Energy Yes

[OPTIONS]
 Units               CMH
 Headloss            H-W
 Specific Gravity    1
 Viscosity           1
 Trials              100
 Accuracy            0.0001
 CHECKFREQ           2
```

```
 MAXCHECK            10
 DAMPLIMIT           0
 Unbalanced          Continue
 Pattern             1
 Demand Multiplier   1.0
 Emitter Exponent    0.5
 Quality             None mg/L
 Diffusivity         1
 Tolerance           0.01
 Hydraulics SAVE singlePumpNet_hyd

[COORDINATES]
;Node              X-Coord          Y-Coord
 node              2707.76          4944.60
 1                 7442.13          7106.48
 source            258.32           4948.34
 tank              4979.22          7534.63

[VERTICES]
;Link              X-Coord          Y-Coord
 pipe              5020.78          7548.48

[LABELS]
;X-Coord            Y-Coord          Label & Anchor Node
 4720.34           8067.80          "Tank"
 -42.37            5508.47          "Source"
 1245.76           5491.53          "Pump"
 7347.46           6932.20          "n2"
 2618.64           4745.76          "n1"
 6127.12           7661.02          "p2"
 3500.00           6457.63          "p1"

[BACKDROP]
 DIMENSIONS      0.00             0.00             10000.00         10000.00
 UNITS           Meters
 FILE
 OFFSET          0.00             0.00

[END]
```

D.2 Van Zyl test network - EPANET input file

```
[TITLE]
Van Zyl test network | adapted by bcoelho | 2015

[JUNCTIONS]
;ID              Elev         Demand       Pattern
 N1              10           0                         ;
 N2              10           0                         ;
 N3              75           0                         ;
 N361            100          0                         ;
 N5              30           50           pattern24    ;
 N6              30           100          pattern24    ;
 N362            100          0                         ;
 N364            100          0                         ;
 N365            100          0                         ;
 N12             100          0                         ;
 N13             100          0                         ;
 N10             100          0                         ;
 N11             100          0                         ;

[RESERVOIRS]
;ID              Head         Pattern
 R1              20                           ;

[TANKS]
;ID              Elevation    InitLevel    MinLevel     MaxLevel     Diameter     MinVol       VolCurve
 T5              80           4.5          0            5            25           0                           ;
 T6              85           9.5          0            10           20           0                           ;

[PIPES]
;ID              Node1            Node2            Length       Diameter     Roughness    MinorLoss    Status
 p1              R1               N1               1            1000         100          0            Open   ;
 p12             N1               N12              1            1000         100          0            Open   ;
 p10             N1               N10              1            1000         100          0            Open   ;
 p11             N11              N2               1            1000         100          0            Open   ;
 p13             N13              N2               1            1000         100          0            Open   ;
 p2              N2               N3               2600         450          100          0            Open   ;
 p3              N3               T5               1000         350          100          0            Open   ;
 p18             N3               N361             1            1000         100          0            Open   ;
 p5              T5               N5               500          300          100          0            Open   ;
 p361            N361             N362             1            1000         100          0            Open   ;
 p4              N365             T6               2000         350          100          0            Open   ;
 p19             N361             N365             1            1000         100          0            CV     ;
 p7              N6               N5               1            200          100          0            Open   ;
 p6              T6               N6               1100         300          100          0            Open   ;
 1               N364             N365             1            1000         100          0            Open   ;

[PUMPS]
;ID              Node1            Node2            Parameters
 Pump2B          N12              N13              HEAD Head1 PATTERN 2B ;
 Pump1A          N10              N11              HEAD Head1 PATTERN 1A ;
 Pump3B          N362             N364             HEAD Head6 ;

[VALVES]
;ID              Node1            Node2            Diameter     Type Setting      MinorLoss

[TAGS]

[DEMANDS]
;Junction        Demand       Pattern          Category

[STATUS]
;ID              Status/Setting

[PATTERNS]
;ID              Multipliers
;water demand pattern
 pattern24       1.71         1.48         1.02         0.73         0.55         0.49
 pattern24       0.55         0.73         1.02         1.36         1.53         1.53
 pattern24       1.36         1.1          0.91         0.76         0.67         0.62
 pattern24       0.62         0.67         0.76         0.91         1.1          1.48
;Energy price pattern
 pumptariff      0.1194       0.1194       0.1194       0.1194       0.1194       0.1194
 pumptariff      0.1194       0.1194       0.1194       0.1194       0.1194       0.1194
 pumptariff      0.1194       0.1194       0.1194       0.1194       0.1194       0.0244
 pumptariff      0.0244       0.0244       0.0244       0.0244       0.0244       0.0244
;Van Zyl pump pattern (run5)
 1A              0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
 1A              0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
 1A              0.0000       0.0000       0.0000       0.0000       0.0000       1.0000
 1A              1.0000       1.0000       1.0000       1.0000       1.0000       0.0000
;Van Zyl pattern (run5)
```

```
 2B               0.0000       0.0000       0.0000       1.0000       1.0000       1.0000
 2B               1.0000       1.0000       1.0000       1.0000       1.0000       1.0000
 2B               1.0000       1.0000       1.0000       0.0000       0.0000       1.0000
 2B               1.0000       1.0000       1.0000       1.0000       1.0000       1.0000
;Van Zyl pattern (run5)
 3B               0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
 3B               0.0000       0.0000       0.0000       0.0000       0.0000       1.0000
 3B               1.0000       1.0000       1.0000       1.0000       1.0000       1.0000
 3B               1.0000       1.0000       1.0000       1.0000       1.0000       1.0000

[CURVES]
;ID               X-Value      Y-Value
;PUMP: EFFICIENCY: pump efficiency curve
 leff             50           78
 leff             107          80
 leff             151          68
 leff             200          60
;PUMP: PUMP: Pump characteristic curve
 Head1            0            100
 Head1            120          90
 Head1            150          83
;PUMP: PUMP: pump characteristic curve
 Head6            0            120
 Head6            90           75
 Head6            150          0

[CONTROLS]
LINK Pump1A 0.0000 AT TIME 0.0000
LINK Pump1A 0.0000 AT TIME 1.0000
LINK Pump1A 0.0000 AT TIME 2.0000
LINK Pump1A 0.0000 AT TIME 3.0000
LINK Pump1A 0.0000 AT TIME 4.0000
LINK Pump1A 0.0000 AT TIME 5.0000
LINK Pump1A 0.0000 AT TIME 6.0000
LINK Pump1A 0.0000 AT TIME 7.0000
LINK Pump1A 0.0000 AT TIME 8.0000
LINK Pump1A 0.0000 AT TIME 9.0000
LINK Pump1A 0.0000 AT TIME 10.0000
LINK Pump1A 0.0000 AT TIME 11.0000
LINK Pump1A 0.0000 AT TIME 12.0000
LINK Pump1A 0.0000 AT TIME 13.0000
LINK Pump1A 0.0000 AT TIME 14.0000
LINK Pump1A 0.0000 AT TIME 15.0000
LINK Pump1A 0.0000 AT TIME 16.0000
LINK Pump1A 1.0000 AT TIME 17.0000
LINK Pump1A 1.0000 AT TIME 18.0000
LINK Pump1A 1.0000 AT TIME 19.0000
LINK Pump1A 1.0000 AT TIME 20.0000
LINK Pump1A 1.0000 AT TIME 21.0000
LINK Pump1A 1.0000 AT TIME 22.0000
LINK Pump1A 0.0000 AT TIME 23.0000

LINK Pump2B 0.0000 AT TIME 0.0000
LINK Pump2B 0.0000 AT TIME 1.0000
LINK Pump2B 0.0000 AT TIME 2.0000
LINK Pump2B 1.0000 AT TIME 3.0000
LINK Pump2B 1.0000 AT TIME 4.0000
LINK Pump2B 1.0000 AT TIME 5.0000
LINK Pump2B 1.0000 AT TIME 6.0000
LINK Pump2B 1.0000 AT TIME 7.0000
LINK Pump2B 1.0000 AT TIME 8.0000
LINK Pump2B 1.0000 AT TIME 9.0000
LINK Pump2B 1.0000 AT TIME 10.0000
LINK Pump2B 1.0000 AT TIME 11.0000
LINK Pump2B 1.0000 AT TIME 12.0000
LINK Pump2B 1.0000 AT TIME 13.0000
LINK Pump2B 1.0000 AT TIME 14.0000
LINK Pump2B 0.0000 AT TIME 15.0000
LINK Pump2B 0.0000 AT TIME 16.0000
LINK Pump2B 1.0000 AT TIME 17.0000
LINK Pump2B 1.0000 AT TIME 18.0000
LINK Pump2B 1.0000 AT TIME 19.0000
LINK Pump2B 1.0000 AT TIME 20.0000
LINK Pump2B 1.0000 AT TIME 21.0000
LINK Pump2B 1.0000 AT TIME 22.0000
LINK Pump2B 1.0000 AT TIME 23.0000

LINK Pump3B 0.0000 AT TIME 0.0000
LINK Pump3B 0.0000 AT TIME 1.0000
LINK Pump3B 0.0000 AT TIME 2.0000
LINK Pump3B 0.0000 AT TIME 3.0000
LINK Pump3B 0.0000 AT TIME 4.0000
LINK Pump3B 0.0000 AT TIME 5.0000
```

```
LINK Pump3B 0.0000 AT TIME 6.0000
LINK Pump3B 0.0000 AT TIME 7.0000
LINK Pump3B 0.0000 AT TIME 8.0000
LINK Pump3B 0.0000 AT TIME 9.0000
LINK Pump3B 0.0000 AT TIME 10.0000
LINK Pump3B 1.0000 AT TIME 11.0000
LINK Pump3B 1.0000 AT TIME 12.0000
LINK Pump3B 1.0000 AT TIME 13.0000
LINK Pump3B 1.0000 AT TIME 14.0000
LINK Pump3B 1.0000 AT TIME 15.0000
LINK Pump3B 1.0000 AT TIME 16.0000
LINK Pump3B 1.0000 AT TIME 17.0000
LINK Pump3B 1.0000 AT TIME 18.0000
LINK Pump3B 1.0000 AT TIME 19.0000
LINK Pump3B 1.0000 AT TIME 20.0000
LINK Pump3B 1.0000 AT TIME 21.0000
LINK Pump3B 1.0000 AT TIME 22.0000
LINK Pump3B 1.0000 AT TIME 23.0000

[RULES]

[ENERGY]
 Global Efficiency   85
 Global Price        1
 Global Pattern      pumptariff
 Demand Charge       0
 Pump  Pump2B          Efficiency leff
 Pump  Pump2B          Price      1
 Pump  Pump2B          Pattern    pumptariff
 Pump  Pump1A          Efficiency leff
 Pump  Pump1A          Price      1
 Pump  Pump1A          Pattern    pumptariff
 Pump  Pump3B          Price      1
 Pump  Pump3B          Pattern    pumptariff

[EMITTERS]
;Junction         Coefficient

[QUALITY]
;Node             InitQual

[SOURCES]
;Node             Type         Quality      Pattern

[REACTIONS]
;Type      Pipe/Tank        Coefficient

[REACTIONS]
 Order Bulk             1
 Order Tank             1
 Order Wall             1
 Global Bulk            0
 Global Wall            0
 Limiting Potential     0
 Roughness Correlation  0

[MIXING]
;Tank             Model

[TIMES]
 Duration            24:00
 Hydraulic Timestep  1:00
 Quality Timestep    0:05
 Pattern Timestep    1:00
 Pattern Start       0:00
 Report Timestep     1:00
 Report Start        0:00
 Start ClockTime     0:00
 Statistic           NONE

[REPORT]
 Energy Yes
 Status              Full
 Summary             No

[OPTIONS]
 Units               LPS
 Headloss            H-W
 Specific Gravity    1.000000
 Viscosity           1.000000
 Trials              100
 Accuracy            0.0001
 CHECKFREQ           2
```

```
 MAXCHECK            10
 DAMPLIMIT           0
 Unbalanced          Continue
 Pattern             SSP2B
 Demand Multiplier   1.0000
 Emitter Exponent    0.5000
 Quality             None mg/L
 Diffusivity         1
 Tolerance           0.01
 Hydraulics SAVE VanZyl_bcoelho_hyd

[COORDINATES]
;Node              X-Coord          Y-Coord
 N1                200.69           4587.16
 N2                1829.13          4552.75
 N3                2930.05          5791.28
 N361              3319.95          6238.53
 N5                5407.11          6548.17
 N6                5407.11          6892.20
 N362              2964.45          6513.76
 N364              3423.17          7064.22
 N365              3778.67          6788.99
 N12               602.06           5022.94
 N13               1370.41          5022.94
 N10               636.47           4151.38
 N11               1404.82          4151.38
 R1                -521.79          4587.16
 T5                4684.63          6009.17
 T6                4489.68          7672.02

[VERTICES]
;Link              X-Coord          Y-Coord

[LABELS]
;X-Coord            Y-Coord          Label & Anchor Node
 17.20             4483.94          "N1"
 189.22            5172.02          "N12"
 1496.56           5194.95          "N13"
 235.09            4151.38          "N10"
 1519.50           4128.44          "N11"
 2586.01           5905.96          "N3"
 3388.76           6227.06          "N361"
 2528.67           6559.63          "N362"
 3388.76           7339.45          "N364"
 3870.41           6834.86          "N365"
 4053.90           8084.86          "Tank B"
 4260.32           6387.61          "Tank A"
 -590.60           4919.72          "R1"
 5418.58           6444.95          "N5"
 5441.51           7155.96          "N6"
 2035.55           4633.03          "N2"
 693.81            4025.23          "Pump 1A"
 670.87            5389.91          "Pump 2B"
 2391.06           7075.69          "Pump 3B"
 344.04            7505.73          "Richmond"
 504.59            7253.44          "WDN"

[BACKDROP]
 DIMENSIONS      0.00             0.00             10000.00         10000.00
 UNITS           None
 FILE
 OFFSET          0.00             0.00

[END]
```

D.3 Richmond network - EPANET input file

```
[TITLE]
Richmond Skelton Water Supply System
For more information on this network, see the file Richmond_readme.text distributed
with this network, or email Kobus van Zyl at jevz@ing.rau.ac.za or kobusvanzyl@mail.com

Adapted by bcoelho | 2015

[JUNCTIONS]
;ID              Elev         Demand       Pattern
 4               183          0                           ;
 9               68.85        0                           ;
 10              166.42       5.68         domestic       ;
 42              60           3.68         domestic       ;
 104             184          0                           ;
 164             65           0                           ;
 175             65           0                           ;
 186             112.18       0                           ;
 197             115          0                           ;
 206             134.99       0                           ;
 249             101          11.3         domestic       ;
 264             184          0                           ;
 284             186          0                           ;
 312             242          2.13         domestic       ;
 320             242          0                           ;
 321             242          0                           ;
 325             242          3.71         domestic       ;
 353             185.78       0                           ;
 364             217          0                           ;
 632             69           0                           ;
 633             69           0                           ;
 634             139.02       0                           ;
 635             139.82       0                           ;
 636             139.82       0                           ;
 637             140          1            domestic       ;
 701             198.33       .11          domestic       ;
 729             202.15       0                           ;
 745             177          1.42         domestic       ;
 753             177          .1           domestic       ;
 766             69.7         0                           ;
 768             69.59        0                           ;
 770             69.39        0                           ;
 771             69           0                           ;
 1125            184          0                           ;
 1302            216.65       16.25        domestic       ;
 1963            70           0                           ;
 2009            70           0                           ;
 1250            186          0                           ;
 774             188          0                           ;
 2010            69           0                           ;
 777             100          -9.16                       ;

[RESERVOIRS]
;ID              Head         Pattern
 0               1            40              ;

[TANKS]
;ID              Elevation    InitLevel    MinLevel     MaxLevel     Diameter     MinVol       VolCurve
 C               258.9        1.84         0            2            6.6          0                           ;
 A               184.13       3.12         0.00         3.37         23.5         0                           ;
 D               241.18       1.94         0.00         2.11         11.8         0                           ;
 B               216          3.37         0.00         3.65         15.4         0                           ;
 E               203.01       2.47         0.00         2.69         8            0                           ;
 F               235.71       1.96         0.00         2.19         3.6          0                           ;

[PIPES]
;ID              Node1            Node2            Length       Diameter     Roughness    MinorLoss    Status
 788             A                4                18           150          120          0            Open   ;
 790             4                10               401          150          120          0            Open   ;
 793             4                104              18           150          120          0            Open   ;
 794             9                42               1087         300          140          0            Open   ;
 841             42               164              3002         250          140          0            Open   ;
 911             104              206              846          152          110          0            Open   ;
 912             104              264              44           100          100          0            Open   ;
 993             164              175              18           250          130          0            Open   ;
 1020            186              197              5            250          140          0            Open   ;
 1033            164              197              16           250          137.5        0            CV     ;
 1036            197              284              2003         250          130          0            Open   ;
 1085            206              249              30           76           80           0            Open   ;
 1107            249              634              6            100          70           0            Open   ;
 1153            284              1250             13           200          130          0            Open   ;
 1154            1125             312              1261         150          100          0            CV     ;
```

```
 1178            284             774             18          229         75          0           Open   ;
 1196            312             D               15          100         30          0           CV     ;
 1208            320             321             7           80          90          0           Open   ;
 1209            320             325             3           150         100         0           Open   ;
 1210            321             312             23          150         100         0           CV     ;
 1278            353             364             472         200         130         0           Open   ;
 1301            B               1302            10          152         70          0           Open   ;
 1304            B               364             7           152         70          0           Open   ;
 1638            632             633             5           300         120         0           Open   ;
 1645            634             635             2           100         130         0           Open   ;
 1653            636             637             1           100         120         0           CV     ;
 1677            2010            770             5           300         120         0           CV     ;
 1740            637             C               4028        100         130         0           Open   ;
 1752            321             701             365         150         140         0           Open   ;
 1753            701             729             5612        152         110         0           Open   ;
 1783            729             E               1591        76          70          0           CV     ;
 1793            E               745             444         102         70          0           CV     ;
 1832            753             F               1216        76          87.5        0           Open   ;
 1842            766             768             5           300         150         0           Open   ;
 1844            768             770             4           300         150         0           Open   ;
 1848            770             771             5           300         150         0           Open   ;
 1849            771             9               26          150         100         0           Open   ;
 1879            774             A               1           150         100         0           Open   ;
 1913            0               632             8           300         120         0           Open   ;
 1964            633             1963            4           150         100         0           Open   ;
 1978            D               320             17          150         100         0           Open   ;
 2010            632             2009            4           150         100         0           Open   ;
 p1              633             2010            1           300         120         0           Open   ;
 p2              777             A               20          200         100         0           Open   ;

[PUMPS]
;ID             Node1           Node2           Parameters
 7F             745             753             HEAD 1883 PATTERN pat7F ;
 2A             1963            768             HEAD 2015 PATTERN pat2A ;
 5C             635             636             HEAD 1884 PATTERN pat5C ;
 6D             264             1125            HEAD 1123 PATTERN pat6D ;
 3A             175             186             HEAD 1006 PATTERN pat3A ;
 4B             1250            353             HEAD 1881 PATTERN pat4B ;
 1A             2009            766             HEAD 2007 PATTERN pat1A ;

[VALVES]
;ID             Node1           Node2           Diameter    Type Setting     MinorLoss

[TAGS]

[DEMANDS]
;Junction       Demand      Pattern         Category

[STATUS]
;ID             Status/Setting
 7F             Closed
 2A             Closed
 5C             Closed
 6D             Closed
 3A             Closed
 4B             Closed
 1A             Closed

[PATTERNS]
;ID             Multipliers
;
 Fac_11         1           1           1           1           1           1
 Fac_11         1           1           1           1           1           1
 Fac_11         1           1           1           1           1           1
 Fac_11         1           1           1           1           1           1
;
 CBTariff       2.40925     2.40925     2.40925     2.40925     2.40925     2.40925
 CBTariff       2.40925     6.7945      6.7945      6.7945      6.7945      6.7945
 CBTariff       6.7945      6.7945      6.7945      6.7945      6.7945      6.7945
 CBTariff       6.7945      6.7945      6.7945      6.7945      6.7945      6.7945
;
 HHTariff       2.46        2.46        2.46        2.46        2.46        2.46
 HHTariff       2.46        9.866       9.866       9.866       9.866       9.866
 HHTariff       9.866       9.866       9.866       9.866       9.866       9.866
 HHTariff       9.866       9.866       9.866       9.866       9.866       9.866
;
 LZGTariff      2.460       2.460       2.460       2.460       2.460       2.460
 LZGTariff      2.460       11.195      11.195      11.195      11.195      11.195
 LZGTariff      11.195      11.195      11.195      11.195      11.195      11.195
 LZGTariff      11.195      11.195      11.195      11.195      11.195      11.195
;
 LZHZTariff     2.456666667 2.456666667 2.456666667 2.456666667 2.456666667 2.456666667
 LZHZTariff     2.456666667 12.34       12.34       12.34       12.34       12.34
```

```
 LZHZTariff       12.34        12.34        12.34        12.34        12.34        12.34
 LZHZTariff       12.34        12.34        12.34        12.34        12.34        12.34
;
 STariff          2.44         2.44         2.44         2.44         2.44         2.44
 STariff          2.44         11.94        11.94        11.94        11.94        11.94
 STariff          11.94        11.94        11.94        11.94        11.94        11.94
 STariff          11.94        11.94        11.94        11.94        11.94        11.94
;
 STTariff         2.41         2.41         2.41         2.41         2.41         2.41
 STTariff         2.41         7.535        7.535        7.535        7.535        7.535
 STTariff         7.535        7.535        7.535        7.535        7.535        7.535
 STTariff         7.535        7.535        7.535        7.535        7.535        7.535
;
 40               70.33        69.55        69.42        69.42        70.33        70.33
 40               70.33        70.33        70.33        70.33        70.29        70.29
 40               70.33        70.42        70.42        70.37        69.64        69.68
 40               69.68        70.42        70.37        70.33        70.33        70.33
;
 domestic         1.10         1.61         1.53         1.4          1.15         1.06
 domestic         1.04         1            .92          .95          1.16         1.34
 domestic         1.45         1.32         1.33         1.11         1.07         .71
 domestic         .48          .46          .4           .39          .41          .52
;speed pattern for pump 1A
 pat1A            0.0000       1.0000       0.0000       0.0000       0.0000       0.0000
 pat1A            0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
 pat1A            0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
 pat1A            0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
;speed pattern for pump 2A
 pat2A            1.0000       1.0000       1.0000       1.0000       1.0000       1.0000
 pat2A            1.0000       1.0000       1.0000       1.0000       1.0000       1.0000
 pat2A            1.0000       1.0000       1.0000       1.0000       1.0000       1.0000
 pat2A            1.0000       1.0000       1.0000       1.0000       1.0000       1.0000
;speed pattern for pump 3A
 pat3A            1.0000       1.0000       1.0000       1.0000       1.0000       1.0000
 pat3A            1.0000       1.0000       1.0000       1.0000       1.0000       1.0000
 pat3A            1.0000       1.0000       1.0000       1.0000       1.0000       1.0000
 pat3A            1.0000       1.0000       1.0000       1.0000       1.0000       1.0000
;speed pattern for pump 4B
 pat4B            0            1            1            1            0            0
 pat4B            1            0            1            0            1            0
 pat4B            1            1            0            1            0            1
 pat4B            0            0            1            0            0            1
;speed pattern for pump 5C
 pat5C            0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
 pat5C            0.0000       1.0000       1.0000       1.0000       0.0000       0.0000
 pat5C            0.0000       0.0000       0.0000       0.0000       1.0000       1.0000
 pat5C            1.0000       0.0000       0.0000       0.0000       0.0000       1.0000
;speed pattern for pump 6D
 pat6D            1.0000       1.0000       1.0000       1.0000       1.0000       1.0000
 pat6D            1.0000       1.0000       1.0000       1.0000       1.0000       1.0000
 pat6D            1.0000       1.0000       1.0000       1.0000       1.0000       1.0000
 pat6D            1.0000       0.0000       1.0000       1.0000       1.0000       1.0000
;speed pattern for pump 7F
 pat7F            0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
 pat7F            1.0000       0.0000       0.0000       0.0000       0.0000       0.0000
 pat7F            0.0000       0.0000       0.0000       0.0000       0.0000       1.0000
 pat7F            0.0000       0.0000       0.0000       0.0000       0.0000       0.0000

[CURVES]
;ID              X-Value      Y-Value
;PUMP: PUMP:
 1006             0            38.00
 1006             10           37.00
 1006             20           36.00
 1006             30           34.00
 1006             40           31.00
 1006             50           29.00
 1006             60           27.00
 1006             70           25.00
;PUMP: PUMP:
 1123             0            88.000
 1123             2.78         87.000
 1123             5.56         84.000
 1123             8.53         76.000
 1123             11.11        63.000
 1123             13.89        47.000
;PUMP: PUMP:
 1641             0            146.000
 1641             10           145.000
 1641             15           144.000
 1641             20           143.000
 1641             25           141.000
 1641             30           138.000
```

```
 1641            35          133.000
 1641            40          127.000
 1641            45          120.000
 1641            50          108.000
;PUMP: PUMP:
 1659            0           122.000
 1659            2.22        120.000
 1659            2.78        118.000
 1659            3.33        115.000
 1659            3.89        110.000
 1659            4.44        105.000
 1659            5           96.000
 1659            5.55        85.000
 1659            6.11        72.000
;PUMP: PUMP:
 1691            0           129
 1691            10          128
 1691            15          127
 1691            20          126
 1691            25          124
 1691            30          121
 1691            35          116
 1691            40          110
 1691            45          103
 1691            50          91
;PUMP: PUMP:
 1881            0           40.00
 1881            6.94        38.50
 1881            13.88       38.00
 1881            20.83       37.50
 1881            27.77       36.00
 1881            34.72       35.00
 1881            41.66       32.50
 1881            48.61       30.00
 1881            55.55       27.50
 1881            111.5       7.50
;PUMP: PUMP:
 1883            0           37.000
 1883            1           36.999
 1883            2           36.998
 1883            3           36.997
 1883            4           36.996
 1883            5           36.995
 1883            6           36.994
;PUMP: PUMP:
 1884            0           122.000
 1884            2.22        120.000
 1884            2.78        118.000
 1884            3.33        115.000
 1884            3.89        110.000
 1884            4.44        105.000
 1884            5           96.000
 1884            5.55        85.000
 1884            6.11        72.000
;PUMP: PUMP:
 1948            0           20.00
 1948            0.4         19.00
 1948            0.8         16.00
 1948            1.2         13.00
 1948            1.6         11.00
 1948            2           7.00
;PUMP: PUMP:
 2007            0           129
 2007            10          128
 2007            15          127
 2007            20          126
 2007            25          124
 2007            30          121
 2007            35          116
 2007            40          110
 2007            45          103
 2007            50          91
;PUMP: PUMP:
 2015            0           129
 2015            10          128
 2015            15          127
 2015            20          126
 2015            25          124
 2015            30          121
 2015            35          116
 2015            40          110
 2015            45          103
 2015            50          91
```

```
;PUMP: EFFICIENCY:
 HHEfficiency     0            0
 HHEfficiency     1.1          0
 HHEfficiency     1.5          48
 HHEfficiency     2            57
 HHEfficiency     2.5          62
 HHEfficiency     3            67
 HHEfficiency     3.5          71
 HHEfficiency     4.5          71
 HHEfficiency     5            69
 HHEfficiency     5.5          67
 HHEfficiency     6            62
;PUMP: EFFICIENCY:
 CBNEfficiency    0            0
 CBNEfficiency    20           57
 CBNEfficiency    25           65
 CBNEfficiency    30           71
 CBNEfficiency    35           75
 CBNEfficiency    40           75
 CBNEfficiency    45           75
 CBNEfficiency    50           73
;PUMP: EFFICIENCY:
 CBOEfficiency    0            0
 CBOEfficiency    20           57
 CBOEfficiency    25           65
 CBOEfficiency    30           71
 CBOEfficiency    35           75
 CBOEfficiency    40           75
 CBOEfficiency    45           72
 CBOEfficiency    50           70
;PUMP: EFFICIENCY:
 LZGEfficiency    0            0
 LZGEfficiency    2.7          32
 LZGEfficiency    5.5          50
 LZGEfficiency    8.3          55
 LZGEfficiency    11.11        58
 LZGEfficiency    13.8         47
;PUMP: EFFICIENCY:
 LZHZEfficiency   0            0
 LZHZEfficiency   10           40
 LZHZEfficiency   20           50
 LZHZEfficiency   30           61
 LZHZEfficiency   40           68
 LZHZEfficiency   50           71
 LZHZEfficiency   60           72
 LZHZEfficiency   70           72
 LZHZEfficiency   80           71
 LZHZEfficiency   90           68
 LZHZEfficiency   100          61
 LZHZEfficiency   110          50
 LZHZEfficiency   120          40
;PUMP: EFFICIENCY:
 SEfficiency      0            0
 SEfficiency      1.38         31
 SEfficiency      2.7          48
 SEfficiency      4.6          54
 SEfficiency      5.5          54
 SEfficiency      6.94         43
;PUMP: EFFICIENCY:
 STEfficiency     0            0
 STEfficiency     20           34
 STEfficiency     40           58
 STEfficiency     60           72
 STEfficiency     80           77

[CONTROLS]
LINK 1A 0.0000 AT TIME 0.0000
LINK 1A 1.0000 AT TIME 1.0000
LINK 1A 0.0000 AT TIME 2.0000
LINK 1A 0.0000 AT TIME 3.0000
LINK 1A 0.0000 AT TIME 4.0000
LINK 1A 0.0000 AT TIME 5.0000
LINK 1A 0.0000 AT TIME 6.0000
LINK 1A 0.0000 AT TIME 7.0000
LINK 1A 0.0000 AT TIME 8.0000
LINK 1A 0.0000 AT TIME 9.0000
LINK 1A 0.0000 AT TIME 10.0000
LINK 1A 0.0000 AT TIME 11.0000
LINK 1A 0.0000 AT TIME 12.0000
LINK 1A 0.0000 AT TIME 13.0000
LINK 1A 0.0000 AT TIME 14.0000
LINK 1A 0.0000 AT TIME 15.0000
LINK 1A 0.0000 AT TIME 16.0000
```

```
LINK 1A 0.0000 AT TIME 17.0000
LINK 1A 0.0000 AT TIME 18.0000
LINK 1A 0.0000 AT TIME 19.0000
LINK 1A 0.0000 AT TIME 20.0000
LINK 1A 0.0000 AT TIME 21.0000
LINK 1A 0.0000 AT TIME 22.0000
LINK 1A 0.0000 AT TIME 23.0000

LINK 2A 1.0000 AT TIME 0.0000
LINK 2A 1.0000 AT TIME 1.0000
LINK 2A 1.0000 AT TIME 2.0000
LINK 2A 1.0000 AT TIME 3.0000
LINK 2A 1.0000 AT TIME 4.0000
LINK 2A 1.0000 AT TIME 5.0000
LINK 2A 1.0000 AT TIME 6.0000
LINK 2A 1.0000 AT TIME 7.0000
LINK 2A 1.0000 AT TIME 8.0000
LINK 2A 1.0000 AT TIME 9.0000
LINK 2A 1.0000 AT TIME 10.0000
LINK 2A 1.0000 AT TIME 11.0000
LINK 2A 1.0000 AT TIME 12.0000
LINK 2A 1.0000 AT TIME 13.0000
LINK 2A 1.0000 AT TIME 14.0000
LINK 2A 1.0000 AT TIME 15.0000
LINK 2A 1.0000 AT TIME 16.0000
LINK 2A 1.0000 AT TIME 17.0000
LINK 2A 1.0000 AT TIME 18.0000
LINK 2A 1.0000 AT TIME 19.0000
LINK 2A 1.0000 AT TIME 20.0000
LINK 2A 1.0000 AT TIME 21.0000
LINK 2A 1.0000 AT TIME 22.0000
LINK 2A 1.0000 AT TIME 23.0000

LINK 3A 1.0000 AT TIME 0.0000
LINK 3A 1.0000 AT TIME 1.0000
LINK 3A 1.0000 AT TIME 2.0000
LINK 3A 1.0000 AT TIME 3.0000
LINK 3A 1.0000 AT TIME 4.0000
LINK 3A 1.0000 AT TIME 5.0000
LINK 3A 1.0000 AT TIME 6.0000
LINK 3A 1.0000 AT TIME 7.0000
LINK 3A 1.0000 AT TIME 8.0000
LINK 3A 1.0000 AT TIME 9.0000
LINK 3A 1.0000 AT TIME 10.0000
LINK 3A 1.0000 AT TIME 11.0000
LINK 3A 1.0000 AT TIME 12.0000
LINK 3A 1.0000 AT TIME 13.0000
LINK 3A 1.0000 AT TIME 14.0000
LINK 3A 1.0000 AT TIME 15.0000
LINK 3A 1.0000 AT TIME 16.0000
LINK 3A 1.0000 AT TIME 17.0000
LINK 3A 1.0000 AT TIME 18.0000
LINK 3A 1.0000 AT TIME 19.0000
LINK 3A 1.0000 AT TIME 20.0000
LINK 3A 1.0000 AT TIME 21.0000
LINK 3A 1.0000 AT TIME 22.0000
LINK 3A 1.0000 AT TIME 23.0000

LINK 4B 0.0000 AT TIME 0.0000
LINK 4B 1.0000 AT TIME 1.0000
LINK 4B 1.0000 AT TIME 2.0000
LINK 4B 1.0000 AT TIME 3.0000
LINK 4B 0.0000 AT TIME 4.0000
LINK 4B 0.0000 AT TIME 5.0000
LINK 4B 1.0000 AT TIME 6.0000
LINK 4B 0.0000 AT TIME 7.0000
LINK 4B 1.0000 AT TIME 8.0000
LINK 4B 0.0000 AT TIME 9.0000
LINK 4B 1.0000 AT TIME 10.0000
LINK 4B 0.0000 AT TIME 11.0000
LINK 4B 1.0000 AT TIME 12.0000
LINK 4B 1.0000 AT TIME 13.0000
LINK 4B 0.0000 AT TIME 14.0000
LINK 4B 1.0000 AT TIME 15.0000
LINK 4B 0.0000 AT TIME 16.0000
LINK 4B 1.0000 AT TIME 17.0000
LINK 4B 0.0000 AT TIME 18.0000
LINK 4B 0.0000 AT TIME 19.0000
LINK 4B 1.0000 AT TIME 20.0000
LINK 4B 0.0000 AT TIME 21.0000
LINK 4B 0.0000 AT TIME 22.0000
LINK 4B 1.0000 AT TIME 23.0000
```

```
LINK 5C 0.0000 AT TIME 0.0000
LINK 5C 0.0000 AT TIME 1.0000
LINK 5C 0.0000 AT TIME 2.0000
LINK 5C 0.0000 AT TIME 3.0000
LINK 5C 0.0000 AT TIME 4.0000
LINK 5C 0.0000 AT TIME 5.0000
LINK 5C 0.0000 AT TIME 6.0000
LINK 5C 1.0000 AT TIME 7.0000
LINK 5C 1.0000 AT TIME 8.0000
LINK 5C 1.0000 AT TIME 9.0000
LINK 5C 0.0000 AT TIME 10.0000
LINK 5C 0.0000 AT TIME 11.0000
LINK 5C 0.0000 AT TIME 12.0000
LINK 5C 0.0000 AT TIME 13.0000
LINK 5C 0.0000 AT TIME 14.0000
LINK 5C 0.0000 AT TIME 15.0000
LINK 5C 1.0000 AT TIME 16.0000
LINK 5C 1.0000 AT TIME 17.0000
LINK 5C 1.0000 AT TIME 18.0000
LINK 5C 0.0000 AT TIME 19.0000
LINK 5C 0.0000 AT TIME 20.0000
LINK 5C 0.0000 AT TIME 21.0000
LINK 5C 0.0000 AT TIME 22.0000
LINK 5C 1.0000 AT TIME 23.0000

LINK 6D 1.0000 AT TIME 0.0000
LINK 6D 1.0000 AT TIME 1.0000
LINK 6D 1.0000 AT TIME 2.0000
LINK 6D 1.0000 AT TIME 3.0000
LINK 6D 1.0000 AT TIME 4.0000
LINK 6D 1.0000 AT TIME 5.0000
LINK 6D 1.0000 AT TIME 6.0000
LINK 6D 1.0000 AT TIME 7.0000
LINK 6D 1.0000 AT TIME 8.0000
LINK 6D 1.0000 AT TIME 9.0000
LINK 6D 1.0000 AT TIME 10.0000
LINK 6D 1.0000 AT TIME 11.0000
LINK 6D 1.0000 AT TIME 12.0000
LINK 6D 1.0000 AT TIME 13.0000
LINK 6D 1.0000 AT TIME 14.0000
LINK 6D 1.0000 AT TIME 15.0000
LINK 6D 1.0000 AT TIME 16.0000
LINK 6D 1.0000 AT TIME 17.0000
LINK 6D 1.0000 AT TIME 18.0000
LINK 6D 0.0000 AT TIME 19.0000
LINK 6D 1.0000 AT TIME 20.0000
LINK 6D 1.0000 AT TIME 21.0000
LINK 6D 1.0000 AT TIME 22.0000
LINK 6D 1.0000 AT TIME 23.0000

LINK 7F 0.0000 AT TIME 0.0000
LINK 7F 0.0000 AT TIME 1.0000
LINK 7F 0.0000 AT TIME 2.0000
LINK 7F 0.0000 AT TIME 3.0000
LINK 7F 0.0000 AT TIME 4.0000
LINK 7F 0.0000 AT TIME 5.0000
LINK 7F 1.0000 AT TIME 6.0000
LINK 7F 0.0000 AT TIME 7.0000
LINK 7F 0.0000 AT TIME 8.0000
LINK 7F 0.0000 AT TIME 9.0000
LINK 7F 0.0000 AT TIME 10.0000
LINK 7F 0.0000 AT TIME 11.0000
LINK 7F 0.0000 AT TIME 12.0000
LINK 7F 0.0000 AT TIME 13.0000
LINK 7F 0.0000 AT TIME 14.0000
LINK 7F 0.0000 AT TIME 15.0000
LINK 7F 0.0000 AT TIME 16.0000
LINK 7F 1.0000 AT TIME 17.0000
LINK 7F 0.0000 AT TIME 18.0000
LINK 7F 0.0000 AT TIME 19.0000
LINK 7F 0.0000 AT TIME 20.0000
LINK 7F 0.0000 AT TIME 21.0000
LINK 7F 0.0000 AT TIME 22.0000
LINK 7F 0.0000 AT TIME 23.0000


[RULES]


[ENERGY]
 Global Efficiency   75
 Global Price        0
 Demand Charge       0
```

```
 Pump  7F               Efficiency SEfficiency
 Pump  7F               Price      1
 Pump  7F               Pattern    STariff
 Pump  2A               Efficiency CBNEfficiency
 Pump  2A               Price      1
 Pump  2A               Pattern    CBTariff
 Pump  5C               Efficiency HHEfficiency
 Pump  5C               Price      1
 Pump  5C               Pattern    HHTariff
 Pump  6D               Efficiency LZGEfficiency
 Pump  6D               Price      1
 Pump  6D               Pattern    LZGTariff
 Pump  3A               Efficiency STEfficiency
 Pump  3A               Price      1
 Pump  3A               Pattern    STTariff
 Pump  4B               Efficiency LZHZEfficiency
 Pump  4B               Price      1
 Pump  4B               Pattern    LZHZTariff
 Pump  1A               Efficiency CBOEfficiency
 Pump  1A               Price      1
 Pump  1A               Pattern    CBTariff

[EMITTERS]
;Junction         Coefficient

[QUALITY]
;Node             InitQual

[SOURCES]
;Node             Type         Quality      Pattern

[REACTIONS]
;Type      Pipe/Tank        Coefficient


[REACTIONS]
 Order Bulk             1
 Order Tank             1
 Order Wall             1
 Global Bulk            0
 Global Wall            0
 Limiting Potential     0
 Roughness Correlation  0

[MIXING]
;Tank             Model

[TIMES]
 Duration            24:00
 Hydraulic Timestep  1:00
 Quality Timestep    0:05
 Pattern Timestep    1:00
 Pattern Start       0:00
 Report Timestep     1:00
 Report Start        0:00
 Start ClockTime     0 am
 Statistic           None

[REPORT]
 Status              Full
 Summary             No
 Page                0
 Energy Yes
 Nodes All
 Links All

[OPTIONS]
 Units               LPS
 Headloss            H-W
 Specific Gravity    1
 Viscosity           1
 Trials              40
 Accuracy            0.001
 CHECKFREQ           2
 MAXCHECK            10
 DAMPLIMIT           0
 Unbalanced          Stop
 Pattern             Fac_11
 Demand Multiplier   1.0
 Emitter Exponent    0.5
 Quality             None mg/L
 Diffusivity         1
 Tolerance           0.01
```

```
 Hydraulics SAVE Richmond_bcoelho_hydfile

[COORDINATES]
;Node              X-Coord          Y-Coord
 4                 417432864.00     501468288.00
 9                 420073728.00     500427808.00
 10                417780512.00     500933632.00
 42                419679296.00     500512320.00
 104               417213856.00     501042112.00
 164               419059488.00     500709536.00
 175               419172160.00     501132128.00
 186               418524192.00     501357504.00
 197               418411488.00     500934912.00
 206               416484544.00     500334944.00
 249               416481344.00     500337760.00
 264               416903936.00     501352000.00
 284               417977728.00     502032384.00
 312               416653568.00     503581920.00
 320               416202816.00     504173568.00
 321               416202816.00     503581920.00
 325               415385760.00     504089056.00
 353               417977728.00     503215680.00
 364               417977728.00     503891840.00
 632               421818368.00     500927904.00
 633               421781120.00     500701184.00
 634               415495264.00     499999680.00
 635               415244906.43     499968684.80
 636               414794132.41     499855991.29
 637               414554658.70     499799644.54
 701               416202816.00     501553440.00
 729               415470304.00     501440736.00
 745               414315200.00     501412576.00
 753               413695360.00     501412576.00
 766               421085888.00     501294144.00
 768               421057696.00     500787040.00
 770               421001344.00     500308064.00
 771               420062560.00     500447648.00
 1125              416751840.00     501745312.00
 1302              419048320.00     504060864.00
 1963              421668448.00     500701184.00
 2009              421818368.00     501237792.00
 1250              417977728.00     502539520.00
 774               417696000.00     502060544.00
 2010              421752960.00     500229280.00
 777               417139840.00     502283968.00
 0                 422555897.68     500687105.90
 C                 413382272.00     499661600.00
 A                 417442432.00     502060544.00
 D                 416625408.00     504201728.00
 B                 417921376.00     504370784.00
 E                 414935008.00     501412576.00
 F                 412991040.00     501384384.00

[VERTICES]
;Link              X-Coord          Y-Coord

[LABELS]
;X-Coord            Y-Coord          Label & Anchor Node

[BACKDROP]
 DIMENSIONS      408638249.60     499335350.40     423020758.40     502165641.60
 UNITS           None
 FILE
 OFFSET          0.00             0.00

[END]
```

D.4 Walski network - EPANET input file

```
[TITLE]
Walski network from (Walski et al., 2001) operations chapter exercices | by bcoelho | 2015

[JUNCTIONS]
;ID              Elev         Demand       Pattern
 J10             369          2            DemandPat        ;
 J9              369          2            DemandPat        ;
 J8              372          8            DemandPat        ;
 J11             372          3            DemandPat        ;
 J3              376          9            DemandPat        ;
 J2              389          0                             ;
 J1              375          0                             ;
 J4              381          3            DemandPat        ;
 J7              384          0                             ;
 J5              396          4            DemandFire       ;
 J6              381          25           DemandPat        ;
 J-PRV-2b        360          0                             ;
 J-PRV-2a        360          0                             ;
 J-PRV-1b        360          0                             ;
 J-PRV-1a        360          0                             ;
 J-PMP-1b        319          0                             ;
 J-PMP-1a        319          0                             ;
 J-PMP-2a        373          0                             ;
 J-PMP-2b        373          0                             ;

[RESERVOIRS]
;ID              Head         Pattern
 NewtownRes      320                          ;
 HighFieldRes.   375                          ;

[TANKS]
;ID              Elevation    InitLevel    MinLevel     MaxLevel     Diameter     MinVol       VolCurve
 CentralTank     384          81           75           87           14           0                            ;

[PIPES]
;ID              Node1            Node2            Length       Diameter     Roughness    MinorLoss    Status
 P6a             J3               J-PRV-2a         190          305          110          0            Open   ;
 P6b             J-PRV-2b         J4               191          305          110          0            Open   ;
 P5              J3               J-PRV-1b         168          254          110          0            Open   ;
 P4              J-PRV-1a         J2               183          254          110          0            Open   ;
 P17             J10              J11              229          152          95           0            Open   ;
 P16             J10              J9               375          152          95           0            Open   ;
 P15             J9               J8               396          152          95           0            Open   ;
 P18             J11              J3               373          203          95           0            Open   ;
 P19             J9               J3               221          152          100          0            Open   ;
 P14             J8               J2               244          203          100          0            Closed ;
 P13             J2               J7               255          203          110          0            Open   ;
 P7              J4               J5               259          305          110          0            Open   ;
 P12             J7               J5               503          254          115          0            Open   ;
 P3              J1               J2               701          305          120          0            Open   ;
 P20             J7               CentralTank      47           102          75           0            Open   ;
 P9              J1               J6               640          305          120          0            Open   ;
 P8              J6               J5               1295         305          110          0            Open   ;
 P1              NewtownRes       J-PMP-1a         37           610          120          0            Open   ;
 P2              J-PMP-1b         J1               133          406          120          0            Open   ;
 P10             HighFieldRes.    J-PMP-2a         15           610          105          0            Open   ;
 P11             J-PMP-2b         J5               76           406          105          0            Open   ;

[PUMPS]
;ID              Node1            Node2            Parameters
 PMP-1           J-PMP-1a         J-PMP-1b         HEAD Head-PMP-1 PATTERN PMP-1-pattern ;
 PMP-2           J-PMP-2a         J-PMP-2b         HEAD Head-PMP-2 PATTERN PMP-2-pattern ;

[VALVES]
;ID              Node1            Node2            Diameter     Type Setting      MinorLoss
 PRV-1           J-PRV-1b         J-PRV-1a         12           PRV  52           0            ;
 PRV-2           J-PRV-2b         J-PRV-2a         12           PRV  52           0            ;

[TAGS]

[DEMANDS]
;Junction        Demand       Pattern          Category

[STATUS]
;ID              Status/Setting
 PMP-2           Closed

[PATTERNS]
;ID              Multipliers
;Demand pattern for all junction nodes
 DemandPat       0.6          0.6          0.6          0.75         0.75         0.75
```

```
 DemandPat        1.2          1.2          1.2          1.1          1.1          1.1
 DemandPat        1.15         1.15         1.15         1.2          1.2          1.2
 DemandPat        1.33         1.33         1.33         0.8          0.8          0.8
;Speed pattern for pump PMP-1
 PMP-1-pattern    1            1            1            1            1            1
 PMP-1-pattern    1            1            1            1            1            1
 PMP-1-pattern    1            1            1            1            1            1
 PMP-1-pattern    1            1            1            1            1            1
;Speed pattern for pump PMP-2
 PMP-2-pattern    1            1            1            1            1            1
 PMP-2-pattern    1            1            1            1            1            1
 PMP-2-pattern    1            1            1            1            1            1
 PMP-2-pattern    1            1            1            1            1            1
;demand pattern considering a fire occurrence at 11 (during 3 hours)
 DemandFire       0.6          0.6          0.6          0.75         0.75         0.75
 DemandFire       1.2          1.2          1.2          1.1          1.1          43.85
 DemandFire       43.9         43.9         1.15         1.2          1.2          1.2
 DemandFire       1.33         1.33         1.33         0.8          0.8          0.8

[CURVES]
;ID               X-Value      Y-Value
;PUMP: Pump PMP-1 characteristic curve H-Q
 Head-PMP-1       0            168
 Head-PMP-1       57           160
 Head-PMP-1       125          146
;PUMP: Pump PMP-2 characteristic curve
 Head-PMP-2       0            98
 Head-PMP-2       95           93
 Head-PMP-2       198          84

[CONTROLS]
LINK PMP-1 1.0000 AT TIME 0.00
LINK PMP-1 1.0000 AT TIME 1.00
LINK PMP-1 1.0000 AT TIME 2.00
LINK PMP-1 1.0000 AT TIME 3.00
LINK PMP-1 1.0000 AT TIME 4.00
LINK PMP-1 0.0000 AT TIME 5.00
LINK PMP-1 0.0000 AT TIME 6.00
LINK PMP-1 0.0000 AT TIME 7.00
LINK PMP-1 0.0000 AT TIME 8.00
LINK PMP-1 0.0000 AT TIME 9.00
LINK PMP-1 0.0000 AT TIME 10.00
LINK PMP-1 1.0000 AT TIME 11.00
LINK PMP-1 1.0000 AT TIME 12.00
LINK PMP-1 1.0000 AT TIME 13.00
LINK PMP-1 1.0000 AT TIME 14.00
LINK PMP-1 1.0000 AT TIME 15.00
LINK PMP-1 1.0000 AT TIME 16.00
LINK PMP-1 1.0000 AT TIME 17.00
LINK PMP-1 1.0000 AT TIME 18.00
LINK PMP-1 1.0000 AT TIME 19.00
LINK PMP-1 1.0000 AT TIME 20.00
LINK PMP-1 0.0000 AT TIME 21.00
LINK PMP-1 0.0000 AT TIME 22.00
LINK PMP-1 0.0000 AT TIME 23.00
LINK PMP-1 0.0000 AT TIME 24.00

LINK PMP-2 0.0000 AT TIME 0.00
LINK PMP-2 0.0000 AT TIME 1.00
LINK PMP-2 0.0000 AT TIME 2.00
LINK PMP-2 0.0000 AT TIME 3.00
LINK PMP-2 0.0000 AT TIME 4.00
LINK PMP-2 0.0000 AT TIME 5.00
LINK PMP-2 1.0000 AT TIME 6.00
LINK PMP-2 1.0000 AT TIME 7.00
LINK PMP-2 1.0000 AT TIME 8.00
LINK PMP-2 1.0000 AT TIME 9.00
LINK PMP-2 1.0000 AT TIME 10.00
LINK PMP-2 1.0000 AT TIME 11.00
LINK PMP-2 0.0000 AT TIME 12.00
LINK PMP-2 0.0000 AT TIME 13.00
LINK PMP-2 0.0000 AT TIME 14.00
LINK PMP-2 0.0000 AT TIME 15.00
LINK PMP-2 0.0000 AT TIME 16.00
LINK PMP-2 0.0000 AT TIME 17.00
LINK PMP-2 0.0000 AT TIME 18.00
LINK PMP-2 0.0000 AT TIME 19.00
LINK PMP-2 0.0000 AT TIME 20.00
LINK PMP-2 0.0000 AT TIME 21.00
LINK PMP-2 0.0000 AT TIME 22.00
LINK PMP-2 0.0000 AT TIME 23.00
LINK PMP-2 0.0000 AT TIME 24.00
```

```
LINK PRV-1 OPEN AT TIME 0.00
LINK PRV-1 OPEN AT TIME 1.00
LINK PRV-1 OPEN AT TIME 2.00
LINK PRV-1 OPEN AT TIME 3.00
LINK PRV-1 OPEN AT TIME 4.00
LINK PRV-1 OPEN AT TIME 5.00
LINK PRV-1 OPEN AT TIME 6.00
LINK PRV-1 OPEN AT TIME 7.00
LINK PRV-1 OPEN AT TIME 8.00
LINK PRV-1 OPEN AT TIME 9.00
LINK PRV-1 OPEN AT TIME 10.00
LINK PRV-1 OPEN AT TIME 11.00
LINK PRV-1 OPEN AT TIME 12.00
LINK PRV-1 OPEN AT TIME 13.00
LINK PRV-1 OPEN AT TIME 14.00
LINK PRV-1 OPEN AT TIME 15.00
LINK PRV-1 OPEN AT TIME 16.00
LINK PRV-1 OPEN AT TIME 17.00
LINK PRV-1 OPEN AT TIME 18.00
LINK PRV-1 OPEN AT TIME 19.00
LINK PRV-1 OPEN AT TIME 20.00
LINK PRV-1 OPEN AT TIME 21.00
LINK PRV-1 OPEN AT TIME 22.00
LINK PRV-1 OPEN AT TIME 23.00

LINK PRV-2 OPEN AT TIME 0.00
LINK PRV-2 OPEN AT TIME 1.00
LINK PRV-2 OPEN AT TIME 2.00
LINK PRV-2 OPEN AT TIME 3.00
LINK PRV-2 OPEN AT TIME 4.00
LINK PRV-2 OPEN AT TIME 5.00
LINK PRV-2 OPEN AT TIME 6.00
LINK PRV-2 OPEN AT TIME 7.00
LINK PRV-2 OPEN AT TIME 8.00
LINK PRV-2 OPEN AT TIME 9.00
LINK PRV-2 OPEN AT TIME 10.00
LINK PRV-2 OPEN AT TIME 11.00
LINK PRV-2 OPEN AT TIME 12.00
LINK PRV-2 OPEN AT TIME 13.00
LINK PRV-2 OPEN AT TIME 14.00
LINK PRV-2 OPEN AT TIME 15.00
LINK PRV-2 OPEN AT TIME 16.00
LINK PRV-2 OPEN AT TIME 17.00
LINK PRV-2 OPEN AT TIME 18.00
LINK PRV-2 OPEN AT TIME 19.00
LINK PRV-2 OPEN AT TIME 20.00
LINK PRV-2 OPEN AT TIME 21.00
LINK PRV-2 OPEN AT TIME 22.00
LINK PRV-2 OPEN AT TIME 23.00

[RULES]

[ENERGY]
 Global Efficiency   75
 Global Price        0
 Demand Charge       0

[EMITTERS]
;Junction         Coefficient

[QUALITY]
;Node             InitQual

[SOURCES]
;Node             Type         Quality      Pattern

[REACTIONS]
;Type      Pipe/Tank        Coefficient


[REACTIONS]
 Order Bulk             1
 Order Tank             1
 Order Wall             1
 Global Bulk            0
 Global Wall            0
 Limiting Potential     0
 Roughness Correlation  0

[MIXING]
;Tank             Model

[TIMES]
```

```
 Duration            24
 Hydraulic Timestep  1:00
 Quality Timestep    0:05
 Pattern Timestep    1:00
 Pattern Start       0:00
 Report Timestep     1:00
 Report Start        0:00
 Start ClockTime     12 am
 Statistic           None

[REPORT]
 Status              Full
 Summary             No
 Page                0
 Energy Yes
 Links All
 Nodes All

[OPTIONS]
 Units               LPS
 Headloss            H-W
 Specific Gravity    1
 Viscosity           1
 Trials              40
 Accuracy            0.001
 CHECKFREQ           2
 MAXCHECK            10
 DAMPLIMIT           0
 Unbalanced          Continue
 Pattern             DemandPat
 Demand Multiplier   1.0
 Emitter Exponent    0.5
 Quality             None mg/L
 Diffusivity         1
 Tolerance           0.01
 Hydraulics SAVE walski_hydfile

[COORDINATES]
;Node              X-Coord          Y-Coord
 J10               -1296.61         8525.42
 J9                635.59           8525.42
 J8                3025.42          8525.42
 J11               -1296.61         6881.36
 J3                635.59           6881.36
 J2                3025.42          6881.36
 J1                5381.36          6881.36
 J4                635.59           5169.49
 J7                3025.42          5220.34
 J5                635.59           3355.93
 J6                5381.36          3355.93
 J-PRV-2b          635.59           5779.66
 J-PRV-2a          635.59           6338.98
 J-PRV-1b          1516.95          6881.36
 J-PRV-1a          2279.66          6881.36
 J-PMP-1b          5855.93          7338.98
 J-PMP-1a          6449.15          7847.46
 J-PMP-2a          -635.59          2355.93
 J-PMP-2b          -25.42           2847.46
 NewtownRes        7008.47          8338.98
 HighFieldRes.     -1245.76         1881.36
 CentralTank       2533.90          5847.46

[VERTICES]
;Link              X-Coord          Y-Coord
 P20               2533.90          5423.73

[LABELS]
;X-Coord            Y-Coord          Label & Anchor Node
 -1703.39          8881.36          "J10"
 533.90            9016.95          "J9"
 3110.17           8864.41          "J8"
 -1771.19          6881.36          "J11"
 296.61            7237.29          "J3"
 3127.12           7254.24          "J2"
 5483.05           6847.46          "J1"
 5500.00           3271.19          "J6"
 618.64            3254.24          "J5"
 211.86            5237.29          "J4"
 6262.71           8288.14          "P1"
 5398.31           7491.53          "P2"
 4025.42           7203.39          "P3"
 2483.05           7220.34          "P4"
 923.73            7220.34          "P5"
```

```
 93.22           6677.97         "P6a"
 127.12          5610.17         "P6b"
 262.71          4288.14         "P7"
 2923.73         3694.92         "P8"
 5516.95         5305.08         "P9"
 -720.34         2203.39         "P10"
 144.07          2932.20         "P11"
 1957.63         4288.14         "P12"
 3110.17         6152.54         "P13"
 3093.22         7847.46         "P14"
 1703.39         8864.41         "P15"
 -635.59         8881.36         "P16"
 -1211.86        7813.56         "P17"
 -652.54         7237.29         "P18"
 -1991.53        1644.07         "High Field Res."
 -1127.12        3033.90         "PMP-2"
 6228.81         7474.58         "PMP-1"
 6567.80         8898.31         "Newtown Res."
 1076.27         5966.10         "Central Tank"
 4110.17         2644.07         "High pressure zone"
 -313.56         9474.58         "Low pressure zone"
 703.39          7813.56         "P19"
 2127.12         5389.83         "P20"
 3177.97         5237.29         "J7"
 1567.80         7406.78         "PRV-1"
 -364.41         6169.49         "PRV-2"

[BACKDROP]
 DIMENSIONS      0.00            0.00            10000.00        10000.00
 UNITS           None
 FILE
 OFFSET          0.00            0.00

[END]
```

D.5 Portuguese network - EPANET input file

```
[TITLE]
Simplified model of a Portuguese network | by B. Coelho | 2013
Contact info: A. Andrade-Campos (gilac@ua.pt) or B. Coelho (bcoelho@ua.pt)

[JUNCTIONS]
;ID              Elev         Demand       Pattern
 D1              200          -1           DemandPat1       ;
 J12             315.6        0                             ;
 J4              215          0                             ;
 J5              215          0                             ;
 J7              215          0                             ;
 J6              215          0                             ;
 J8              215          0                             ;
 J9              215          0                             ;
 J15             277.88       0                             ;
 J17             283.88       0                             ;
 J19             274.11       0                             ;
 D5              265.83       1            DemandPat5       ;
 D6              295.39       1            DemandPat6       ;
 J24             241.81       0                             ;
 J20             266.92       0                             ;
 D7              264          1            DemandPat7       ;
 J22             240.11       0                             ;
 J21             227.53       0                             ;
 J29             280          0                             ;
 J30             280          0                             ;
 J32             280          0                             ;
 J31             280          0                             ;
 J33             280          0                             ;
 J34             280          0                             ;
 D8              225          1            DemandPat8       ;
 D9              230          1            DemandPat9       ;
 D11             380          1            DemandPat11      ;
 J37             395.6        0                             ;
 D10             270          1            DemandPat10      ;
 D4              250          1            DemandPat4       ;
 D3              300          1            DemandPat3       ;
 J18             281.25       0                             ;
 J27             283.7        0                             ;
 J1              218.8        0                             ;
 J3              215          0                             ;
 J11             315.6        0                             ;
 J13             311          0                             ;
 J26             283.7        0                             ;
 J28             280          0                             ;
 J36             395.6        0                             ;
 J38             390          0                             ;
 J16             268.55       0                             ;
 J2              218.8        0                             ;
 D2              200          1            DemandPat2       ;
 J25             241.81       0                             ;
 J14             311          0                             ;
 J10             215          0                             ;
 J23             240.11       0                             ;
 J35             0            0                             ;
 J39             390          0                             ;

[RESERVOIRS]
;ID              Head         Pattern

[TANKS]
;ID              Elevation    InitLevel    MinLevel     MaxLevel     Diameter     MinVol       VolCurve
 TankA           215          1.13         0.9          4            36.769553    0                             ;
 TankB           311          3.32         1.5          4.6          22.627417    0                             ;
 TankC           280          3.02         1            3.7          33.941125    0                             ;
 TankD           390          3.94         1.8          5.6          18           0                             ;

[PIPES]
;ID              Node1            Node2            Length       Diameter     Roughness    MinorLoss    Status
 P9              J12              TankB            0.1          1000         130          0            Open   ;
 P4              J4               J5               2            600          130          0            Open   ;
 P5              J4               J7               1            600          130          0            Open   ;
 P29             J29              J30              1            450          130          0            Open   ;
 P30             J29              J32              1            450          130          0            Open   ;
 P31             J31              J34              1            450          140          0            Open   ;
 P32             J33              J34              1            450          140          0            Open   ;
 P34             J37              TankD            0.1          1000         140          0            Open   ;
 P12             J15              D4               100          700          130          0            Open   ;
 CV-A2           TankA            J4               1            600          130          0            CV     ;
 CV-C2           TankC            J29              1            450          130          0            CV     ;
 P16             J17              J18              146.7        500          130          0            Open   ;
```

```
 P17             J18             J19             158.3       500         130         0           Open   ;
 P19             J19             J20             116         500         130         0           Open   ;
 P18             J19             D6              100         500         130         0           Open   ;
 P20             J20             D7              100         500         130         0           Open   ;
 P27             J27             TankC           0.1         1000        130         0           Open   ;
 P0              D1              J1              100         600         130         0           Open   ;
 CV-A1           TankA           J3              1           600         130         0           CV     ;
 CV-B1           TankB           J13             1           700         130         0           CV     ;
 P10             J13             D3              100         700         130         0           Open   ;
 CV-C1           TankC           J28             1           450         130         0           CV     ;
 CV-D            TankD           J38             1           450         140         0           CV     ;
 P13             J15             J16             2896.7      700         130         0           Open   ;
 P21             J20             J21             1918        500         130         0           Open   ;
 P1              J2              TankA           1           600         140         0           Open   ;
 P14             J16             D5              90          400         130         0           Open   ;
 P15             J16             J17             273.7       700         130         0           Open   ;
 P23             J21             J22             566.6       500         130         0           Open   ;
 P22             J21             D8              100         500         130         0           Open   ;
 P24             J22             D9              100         500         130         0           Open   ;
 P3              J3              D2              100         600         130         0           Open   ;
 P26             J25             J26             2096.3      400         130         0           Open   ;
 CV-B2           TankB           J14             1           700         130         0           CV     ;
 P11             J14             J15             2347.26     700         130         0           Open   ;
 P6              J6              J9              2           600         130         0           Open   ;
 P7              J8              J9              1           600         130         0           Open   ;
 P8              J10             J11             2096.98     600         130         0           Open   ;
 P25             J23             J24             186.7       500         130         0           Open   ;
 P33             J35             J36             3100.7      450         140         0           Open   ;
 P35             J39             D11             100         450         140         0           Open   ;
 P28             J28             D10             100         450         130         0           Open   ;

[PUMPS]
;ID             Node1           Node2           Parameters
 PMP-AB1         J5              J6              HEAD CB_PMP-AB PATTERN Pat_PMP-AB1 ;
 PMP-AB2         J7              J8              HEAD CB_PMP-AB PATTERN Pat_PMP-AB2 ;
 PMP-CD1         J30             J31             HEAD CB_PMP-CD PATTERN Pat_PMP-CD1 ;
 PMP-CD2         J32             J33             HEAD CB_PMP-CD PATTERN Pat_PMP-CD2 ;

[VALVES]
;ID             Node1           Node2           Diameter    Type Setting     MinorLoss
 PSV_B           J11             J12             600         PSV  0           0           ;
 PSV_C           J26             J27             600         PSV  0           0           ;
 PSV_D           J36             J37             450         PSV  0           0           ;
 PSV_A           J1              J2              600         PSV  0           0           ;
 ValveG          J24             J25             400         TCV  0           0           ;
 9               J9              J10             600         TCV  450         0           ;
 11              J22             J23             500         TCV  1050        0           ;
 13              J35             J34             450         TCV  100         0           ;
 VB-D            J38             J39             450         TCV  200         0           ;

[TAGS]

[DEMANDS]
;Junction       Demand      Pattern         Category

[STATUS]
;ID             Status/Setting
 PMP-AB2         Closed
 PMP-CD2         Closed

[PATTERNS]
;ID             Multipliers
;PMP-AB1 speed pattern
 Pat_PMP-AB1     1.0000      1.0000      1.0000      0.0000      0.0000      1.0000
 Pat_PMP-AB1     0.0000      1.0000      1.0000      1.0000      0.0000      0.0000
 Pat_PMP-AB1     1.0000      1.0000      1.0000      1.0000      1.0000      1.0000
 Pat_PMP-AB1     1.0000      1.0000      1.0000      1.0000      1.0000      1.0000
;PMP-AB2 speed pattern
 Pat_PMP-AB2     0           0           0           0           0           0
 Pat_PMP-AB2     0           0           0           0           0           0
 Pat_PMP-AB2     0           0           0           0           0           0
 Pat_PMP-AB2     0           0           0           0           0           0
;PMP-CD1 speed pattern
 Pat_PMP-CD1     1.0000      1.0000      1.0000      0.0000      0.0000      0.0000
 Pat_PMP-CD1     0.0000      1.0000      1.0000      0.0000      0.0000      0.0000
 Pat_PMP-CD1     1.0000      1.0000      1.0000      1.0000      1.0000      1.0000
 Pat_PMP-CD1     1.0000      1.0000      1.0000      1.0000      1.0000      1.0000
;PMP-CD2 speed pattern
 Pat_PMP-CD2     0           0           0           0           0           0
 Pat_PMP-CD2     0           0           0           0           0           0
 Pat_PMP-CD2     0           0           0           0           0           0
 Pat_PMP-CD2     0           0           0           0           0           0
;Winter tariff monday to friday
```

```
 tarifaInverno1   0.0843875    0.0843875    0.0831875    0.0831875    0.0831875    0.0831875
 tarifaInverno1   0.0843875    0.1042875    0.1042875    0.1102875    0.1102875    0.1102875
 tarifaInverno1   0.1042875    0.1042875    0.1042875    0.1042875    0.1042875    0.1042875
 tarifaInverno1   0.1102875    0.1102875    0.1102875    0.1042875    0.1042875    0.1042875
;Summer tariff monday to friday
 tarifaVerao1     0.0844875    0.0844875    0.0835875    0.0835875    0.0835875    0.0835875
 tarifaVerao1     0.0844875    0.1043875    0.1043875    0.1100875    0.1100875    0.1100875
 tarifaVerao1     0.1043875    0.1043875    0.1043875    0.1043875    0.1043875    0.1043875
 tarifaVerao1     0.1043875    0.1043875    0.1043875    0.1043875    0.1043875    0.1043875
;Padrao Consumo
 DemandPat1       815.3200     934.3200     937.6800     946.3200     966.3200     659.2400
 DemandPat1       277.1200     493.9200     769.3200     166.4400     0.0000       0.0000
 DemandPat1       477.0800     467.9200     399.8400     745.3200     906.8400     914.3200
 DemandPat1       922.0000     927.1600     924.8400     909.6800     906.3200     808.6800
;Padrao Consumo
 DemandPat2       9.1844       7.5252       6.7404       5.9652       5.8742       6.2360
 DemandPat2       6.7596       9.8456       13.8248      20.4350      19.8844      20.2606
 DemandPat2       20.7446      19.5054      17.5596      17.0756      15.7848      15.7344
 DemandPat2       16.6354      16.5546      16.3952      13.9252      12.1344      11.2854
;Padrao Consumo
 DemandPat3       26.4250      21.7100      18.6050      17.1050      17.5000      17.3950
 DemandPat3       19.0400      27.2250      42.9250      56.8700      59.6450      64.2500
 DemandPat3       64.2500      63.7750      57.2500      50.5650      55.4750      54.7500
 DemandPat3       58.8950      55.0900      47.7500      42.3800      35.5650      32.3100
;Padrao Consumo
 DemandPat4       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
 DemandPat4       0.0000       0.0000       0.0000       0.0000       0.6450       43.1050
 DemandPat4       77.0400      75.0500      5.1600       0.0000       0.0000       0.0000
 DemandPat4       0.0000       13.3650      76.8200      79.6000      3.4650       0.0000
;Padrao Consumo
 DemandPat5       23.4254      21.3000      20.3080      20.0138      19.8468      19.9264
 DemandPat5       20.7566      24.0086      28.1218      32.2632      74.6040      96.1566
 DemandPat5       91.5206      86.8514      86.1018      91.7196      88.5278      87.7978
 DemandPat5       56.6702      31.6842      31.1940      29.8852      26.9676      25.3086
;Padrao Consumo
 DemandPat6       17.2504      12.1196      9.1954       8.3342       7.3254       7.6450
 DemandPat6       9.4100       17.2500      30.9500      43.9796      46.0504      45.7204
 DemandPat6       46.5296      44.6804      40.2350      38.6450      39.1892      37.2900
 DemandPat6       36.8950      36.7708      36.4496      29.0500      24.1746      20.8604
;Padrao Consumo
 DemandPat7       174.2400     166.0000     169.0000     174.1600     177.0000     175.4200
 DemandPat7       174.0000     168.4200     172.5800     168.0000     168.4200     171.4800
 DemandPat7       212.5800     211.0000     203.4200     204.0000     208.0000     204.0000
 DemandPat7       206.5800     205.4200     203.0000     206.5800     284.2800     315.4200
;Padrao Consumo
 DemandPat8       0.0100       0.0100       0.0000       0.0100       0.0000       0.0100
 DemandPat8       0.0100       0.0058       0.0042       0.0100       0.0158       0.0142
 DemandPat8       0.0000       0.0100       0.0100       0.0158       0.0042       0.0100
 DemandPat8       0.0000       0.0100       0.0100       0.0000       0.0100       0.0100
;Padrao Consumo
 DemandPat9       10.4450      10.6416      10.5884      10.5338      18.5012      17.8550
 DemandPat9       11.6858      10.4326      10.4200      10.4316      14.2362      20.7496
 DemandPat9       17.6386      13.3238      19.6154      22.9866      16.8256      15.0304
 DemandPat9       12.3912      17.2268      20.2056      15.4842      13.0910      11.2800
;Padrao Consumo
 DemandPat10      1.5558       0.3332       0.2468       0.1558       0.1358       0.1858
 DemandPat10      0.6022       1.2512       2.7432       5.0854       5.3230       5.1264
 DemandPat10      5.7834       5.5392       4.7950       4.0338       5.6696       3.9378
 DemandPat10      5.0670       4.6318       3.8806       3.4608       2.0726       2.1338
;Padrao Consumo
 DemandPat11      318.5448     320.3054     320.3800     176.7698     0.0000       11.4550
 DemandPat11      311.8804     326.2294     286.2900     98.8952      0.0000       0.0000
 DemandPat11      199.3702     326.4046     326.2500     326.4106     326.6248     326.4342
 DemandPat11      326.3152     326.6848     326.5406     326.7746     326.6848     327.8856
;Summer tariff saturday
 tarifaVerao2     0.0844875    0.0844875    0.0835875    0.0835875    0.0835875    0.0835875
 tarifaVerao2     0.0844875    0.0844875    0.0844875    0.1043875    0.1043875    0.1043875
 tarifaVerao2     0.1043875    0.1043875    0.0844875    0.0844875    0.0844875    0.0844875
 tarifaVerao2     0.0844875    0.0844875    0.1043875    0.1043875    0.0844875    0.0844875
;Summer tariff sunday and national holidays
 tarifaVerao3     0.0844875    0.0844875    0.0835875    0.0835875    0.0835875    0.0835875
 tarifaVerao3     0.0844875    0.0844875    0.0844875    0.0844875    0.0844875    0.0844875
 tarifaVerao3     0.0844875    0.0844875    0.0844875    0.0844875    0.0844875    0.0844875
 tarifaVerao3     0.0844875    0.0844875    0.0844875    0.0844875    0.0844875    0.0844875
;Winter tariff saturday
 tarifaInverno2   0.0843875    0.0843875    0.0831875    0.0831875    0.0831875    0.0831875
 tarifaInverno2   0.0843875    0.0843875    0.0843875    0.1042875    0.1042875    0.1042875
 tarifaInverno2   0.1042875    0.0843875    0.0843875    0.0843875    0.0843875    0.0843875
 tarifaInverno2   0.1042875    0.1042875    0.1042875    0.1042875    0.0843875    0.0843875
;Winter tariff sunday and national holidays
 tarifaInverno3   0.0843875    0.084387     0.0831875    0.0831875    0.0831875    0.0831875
 tarifaInverno3   0.084387     0.084387     0.084387     0.084387     0.084387     0.084387
 tarifaInverno3   0.084387     0.084387     0.084387     0.084387     0.084387     0.084387
```

```
 tarifaInverno3   0.084387     0.084387     0.084387     0.084387     0.084387     0.084387

[CURVES]
;ID               X-Value      Y-Value
;PUMP: PUMP: PUMP: PUMP: PUMP: PUMP: PUMP: PMP-AB characteristic curve
 CB_PMP-AB        0            130
 CB_PMP-AB        943.2        114.5
 CB_PMP-AB        2340         0
;PUMP: PUMP: PUMP: PUMP: PUMP: PUMP: PUMP: PMP-AB efficiency curve
 CE_PMP-AB        0            0
 CE_PMP-AB        180          33
 CE_PMP-AB        306          50
 CE_PMP-AB        576          70
 CE_PMP-AB        943.2        82.1
 CE_PMP-AB        1188         84
 CE_PMP-AB        1512         80
 CE_PMP-AB        1656         76
;PUMP: PUMP: PUMP: PUMP: PUMP: PUMP: PUMP: PMP-CD characteristic curve
 CB_PMP-CD        0            156
 CB_PMP-CD        324          127
 CB_PMP-CD        648          0
;PUMP: PUMP: PUMP: PUMP: PUMP: PUMP: PUMP: PMP-CD efficiency curve
 CE_PMP-CD        0            0
 CE_PMP-CD        45           30
 CE_PMP-CD        90           50
 CE_PMP-CD        144          64
 CE_PMP-CD        180          70
 CE_PMP-CD        216          75
 CE_PMP-CD        288          79
 CE_PMP-CD        324          79.6
 CE_PMP-CD        396          79
 CE_PMP-CD        450          75

[CONTROLS]
LINK PMP-AB1 1.0000 AT TIME 0.0000
LINK PMP-AB1 1.0000 AT TIME 1.0000
LINK PMP-AB1 1.0000 AT TIME 2.0000
LINK PMP-AB1 0.0000 AT TIME 3.0000
LINK PMP-AB1 0.0000 AT TIME 4.0000
LINK PMP-AB1 1.0000 AT TIME 5.0000
LINK PMP-AB1 0.0000 AT TIME 6.0000
LINK PMP-AB1 1.0000 AT TIME 7.0000
LINK PMP-AB1 1.0000 AT TIME 8.0000
LINK PMP-AB1 1.0000 AT TIME 9.0000
LINK PMP-AB1 0.0000 AT TIME 10.0000
LINK PMP-AB1 0.0000 AT TIME 11.0000
LINK PMP-AB1 1.0000 AT TIME 12.0000
LINK PMP-AB1 1.0000 AT TIME 13.0000
LINK PMP-AB1 1.0000 AT TIME 14.0000
LINK PMP-AB1 1.0000 AT TIME 15.0000
LINK PMP-AB1 1.0000 AT TIME 16.0000
LINK PMP-AB1 1.0000 AT TIME 17.0000
LINK PMP-AB1 1.0000 AT TIME 18.0000
LINK PMP-AB1 1.0000 AT TIME 19.0000
LINK PMP-AB1 1.0000 AT TIME 20.0000
LINK PMP-AB1 1.0000 AT TIME 21.0000
LINK PMP-AB1 1.0000 AT TIME 22.0000
LINK PMP-AB1 1.0000 AT TIME 24.0000

LINK PMP-AB2 0.0000 AT TIME 0.0000
LINK PMP-AB2 0.0000 AT TIME 1.0000
LINK PMP-AB2 0.0000 AT TIME 2.0000
LINK PMP-AB2 0.0000 AT TIME 3.0000
LINK PMP-AB2 0.0000 AT TIME 4.0000
LINK PMP-AB2 0.0000 AT TIME 5.0000
LINK PMP-AB2 0.0000 AT TIME 6.0000
LINK PMP-AB2 0.0000 AT TIME 7.0000
LINK PMP-AB2 0.0000 AT TIME 8.0000
LINK PMP-AB2 0.0000 AT TIME 9.0000
LINK PMP-AB2 0.0000 AT TIME 10.0000
LINK PMP-AB2 0.0000 AT TIME 11.0000
LINK PMP-AB2 0.0000 AT TIME 12.0000
LINK PMP-AB2 0.0000 AT TIME 13.0000
LINK PMP-AB2 0.0000 AT TIME 14.0000
LINK PMP-AB2 0.0000 AT TIME 15.0000
LINK PMP-AB2 0.0000 AT TIME 16.0000
LINK PMP-AB2 0.0000 AT TIME 17.0000
LINK PMP-AB2 0.0000 AT TIME 18.0000
LINK PMP-AB2 0.0000 AT TIME 19.0000
LINK PMP-AB2 0.0000 AT TIME 20.0000
LINK PMP-AB2 0.0000 AT TIME 21.0000
LINK PMP-AB2 0.0000 AT TIME 22.0000
LINK PMP-AB2 0.0000 AT TIME 23.0000
```

```
LINK PMP-CD1 1.0000 AT TIME 0.0000
LINK PMP-CD1 1.0000 AT TIME 1.0000
LINK PMP-CD1 1.0000 AT TIME 2.0000
LINK PMP-CD1 0.0000 AT TIME 3.0000
LINK PMP-CD1 0.0000 AT TIME 4.0000
LINK PMP-CD1 0.0000 AT TIME 5.0000
LINK PMP-CD1 0.0000 AT TIME 6.0000
LINK PMP-CD1 1.0000 AT TIME 7.0000
LINK PMP-CD1 1.0000 AT TIME 8.0000
LINK PMP-CD1 0.0000 AT TIME 9.0000
LINK PMP-CD1 0.0000 AT TIME 10.0000
LINK PMP-CD1 0.0000 AT TIME 11.0000
LINK PMP-CD1 1.0000 AT TIME 12.0000
LINK PMP-CD1 1.0000 AT TIME 13.0000
LINK PMP-CD1 1.0000 AT TIME 14.0000
LINK PMP-CD1 1.0000 AT TIME 15.0000
LINK PMP-CD1 1.0000 AT TIME 16.0000
LINK PMP-CD1 1.0000 AT TIME 17.0000
LINK PMP-CD1 1.0000 AT TIME 18.0000
LINK PMP-CD1 1.0000 AT TIME 19.0000
LINK PMP-CD1 1.0000 AT TIME 20.0000
LINK PMP-CD1 1.0000 AT TIME 21.0000
LINK PMP-CD1 1.0000 AT TIME 22.0000
LINK PMP-CD1 1.0000 AT TIME 23.0000

LINK PMP-CD2 0.0000 AT TIME 0.0000
LINK PMP-CD2 0.0000 AT TIME 1.0000
LINK PMP-CD2 0.0000 AT TIME 2.0000
LINK PMP-CD2 0.0000 AT TIME 3.0000
LINK PMP-CD2 0.0000 AT TIME 4.0000
LINK PMP-CD2 0.0000 AT TIME 5.0000
LINK PMP-CD2 0.0000 AT TIME 6.0000
LINK PMP-CD2 0.0000 AT TIME 7.0000
LINK PMP-CD2 0.0000 AT TIME 8.0000
LINK PMP-CD2 0.0000 AT TIME 9.0000
LINK PMP-CD2 0.0000 AT TIME 10.0000
LINK PMP-CD2 0.0000 AT TIME 11.0000
LINK PMP-CD2 0.0000 AT TIME 12.0000
LINK PMP-CD2 0.0000 AT TIME 13.0000
LINK PMP-CD2 0.0000 AT TIME 14.0000
LINK PMP-CD2 0.0000 AT TIME 15.0000
LINK PMP-CD2 0.0000 AT TIME 16.0000
LINK PMP-CD2 0.0000 AT TIME 17.0000
LINK PMP-CD2 0.0000 AT TIME 18.0000
LINK PMP-CD2 0.0000 AT TIME 19.0000
LINK PMP-CD2 0.0000 AT TIME 20.0000
LINK PMP-CD2 0.0000 AT TIME 21.0000
LINK PMP-CD2 0.0000 AT TIME 22.0000
LINK PMP-CD2 0.0000 AT TIME 23.0000

LINK ValveG OPEN AT TIME 0.0000
LINK ValveG CLOSED AT TIME 1.8000
LINK ValveG OPEN AT TIME 2.0000
LINK ValveG CLOSED AT TIME 3.5000
LINK ValveG CLOSED AT TIME 4.5000
LINK ValveG CLOSED AT TIME 5.0000
LINK ValveG CLOSED AT TIME 6.0000
LINK ValveG OPEN AT TIME 7.0000
LINK ValveG OPEN AT TIME 8.0000
LINK ValveG OPEN AT TIME 9.0000
LINK ValveG CLOSED AT TIME 10.0000
LINK ValveG OPEN AT TIME 11.0000
LINK ValveG OPEN AT TIME 12.0000
LINK ValveG OPEN AT TIME 13.0000
LINK ValveG OPEN AT TIME 14.0000
LINK ValveG OPEN AT TIME 15.0000
LINK ValveG OPEN AT TIME 16.0000
LINK ValveG OPEN AT TIME 17.0000
LINK ValveG OPEN AT TIME 18.0000
LINK ValveG OPEN AT TIME 19.0000
LINK ValveG OPEN AT TIME 20.0000
LINK ValveG OPEN AT TIME 21.0000
LINK ValveG OPEN AT TIME 22.0000
LINK ValveG OPEN AT TIME 23.0000

[RULES]

[ENERGY]
 Global Efficiency   75
 Global Price        0
 Demand Charge       0
 Pump  PMP-AB1          Efficiency CE_PMP-AB
```

```
 Pump  PMP-AB1          Price      1
 Pump  PMP-AB1          Pattern    tarifaInverno2
 Pump  PMP-AB2          Efficiency CE_PMP-AB
 Pump  PMP-AB2          Price      1
 Pump  PMP-AB2          Pattern    tarifaInverno2
 Pump  PMP-CD1          Efficiency CE_PMP-CD
 Pump  PMP-CD1          Price      1
 Pump  PMP-CD1          Pattern    tarifaInverno2
 Pump  PMP-CD2          Efficiency CE_PMP-CD
 Pump  PMP-CD2          Price      1
 Pump  PMP-CD2          Pattern    tarifaInverno2

[EMITTERS]
;Junction         Coefficient

[QUALITY]
;Node             InitQual

[SOURCES]
;Node             Type         Quality      Pattern

[REACTIONS]
;Type      Pipe/Tank        Coefficient


[REACTIONS]
 Order Bulk             1
 Order Tank             1
 Order Wall             1
 Global Bulk            0
 Global Wall            0
 Limiting Potential     0
 Roughness Correlation  0

[MIXING]
;Tank             Model

[TIMES]
 Duration            24:00
 Hydraulic Timestep  1:00
 Quality Timestep    1:00
 Pattern Timestep    1:00
 Pattern Start       0:00
 Report Timestep     1:00
 Report Start        0:00
 Start ClockTime     0 am
 Statistic           NONE

[REPORT]
 Status              Full
 Summary             Yes
 Page                0
 Energy Yes

[OPTIONS]
 Units               CMH
 Headloss            H-W
 Specific Gravity    1
 Viscosity           1
 Trials              50
 Accuracy            0.0001
 CHECKFREQ           2
 MAXCHECK            10
 DAMPLIMIT           0
 Unbalanced          Continue 10
 Pattern             Pat_PMP
 Demand Multiplier   1.0
 Emitter Exponent    0.5
 Quality             None mg/L
 Diffusivity         1
 Tolerance           0.01
 Hydraulics SAVE portugueseNet_hyd

[COORDINATES]
;Node             X-Coord          Y-Coord
 D1                4347.46          -2677.97
 J12               8962.16          -1043.58
 J4                6209.86          -3887.61
 J5                6565.37          -3302.75
 J7                6550.85          -4491.53
 J6                7333.72          -3291.28
 J8                7347.46          -4491.53
 J9                7723.62          -3887.61
```

```
 J15              10228.81         -2169.49
 J17              11279.66         -2050.85
 J19              12144.07         -2423.73
 D5               10805.08         -3474.58
 D6               12144.07         -1355.93
 J24              15177.97         -3423.73
 J20              12635.59         -2898.31
 D7               12635.59         -4830.51
 J22              13923.73         -3661.02
 J21              13279.66         -3966.10
 J29              17500.00         -2644.07
 J30              17754.24         -2118.64
 J32              17771.19         -3118.64
 J31              18432.20         -2118.64
 J33              18500.00         -3118.64
 J34              18720.34         -2627.12
 D8               13279.66         -5372.88
 D9               13923.73         -5016.95
 D11              20110.17         -3915.25
 J37              19855.93         -762.71
 D10              16923.73         -5067.80
 D4               10228.81         -4237.29
 D3               9228.81          -3101.69
 J18              11652.54         -2610.17
 J27              16567.80         -2220.34
 J1               4340.60          -3704.13
 J3               5452.98          -4862.39
 J11              8415.25          -1050.85
 J13              9228.81          -2033.90
 J26              16042.37         -2220.34
 J28              16923.73         -3372.88
 J36              19381.36         -762.71
 J38              20110.17         -1847.46
 J16              10805.08         -2661.02
 J2               5120.41          -3692.66
 D2               5452.98          -5802.75
 J25              15754.24         -3423.73
 J14              9974.58          -1288.14
 J10              8161.02          -3881.36
 J23              14584.75         -3661.02
 J35              19194.92         -2627.12
 J39              20110.17         -2491.53
 TankA            5470.18          -3864.68
 TankB            9225.92          -1295.87
 TankC            16923.73         -2644.07
 TankD            20127.12         -1067.80

[VERTICES]
;Link             X-Coord          Y-Coord

[LABELS]
;X-Coord           Y-Coord          Label & Anchor Node
 4262.71          -2237.29         "D1"
 5177.97          -3118.64         "TankA"
 8957.63          -593.22          "TankB"
 5330.51          -5983.05         "D2"
 12025.42         -966.10          "D6"
 16635.59         -1423.73         "TankC"
 12516.95         -5000.00         "D7"
 13161.02         -5525.42         "D8"
 13788.14         -5169.49         "D9"
 19822.03         -338.98          "TankD"
 16754.24         -5254.24         "D10"
 10652.54         -3627.12         "D5"
 10076.27         -4406.78         "D4"
 9110.17          -3271.19         "D3"
 28974.36         -7741.62         "DemandPat11"
 15364.41         -2898.31         "G"
 19940.68         -4084.75         "D11"
 6466.10          -2864.41         "PMP-AB1"
 6449.15          -4627.12         "PMP-AB2"
 17601.69         -1677.97         "PMP-CD1"
 17669.49         -3254.24         "PMP-CD2"
 20279.66         -2033.90         "VB-D"

[BACKDROP]
 DIMENSIONS       0.00             0.00             10000.00         10000.00
 UNITS            None
 FILE
 OFFSET           0.00             0.00

[END]
```

E ANN-based models forecasting results

Table E.1: Training forecast accuracy measures obtained for each data set with the distinct ANN-based models developed.

ANN model	R^2 (-)	NSE (-)	MAE (m^3/h)	RMSE (m^3/h)	MAPE (%)	maxAE (m^3/h)
WD2_hist	0.76	0.76	1.72	2.24	1.97E+13	14.59
WD2_1lag	0.94	0.94	0.85	1.16	1.26E+13	9.06
WD2_3lags	0.96	0.96	0.71	0.96	1.32E+13	8.87
WD2_antrhop	0.93	0.93	0.84	1.24	1.56E+13	13.22
WD2_neighb	0.96	0.96	0.69	0.96	9.41E+12	10.32
WD2_meteo	0.87	0.87	1.24	1.68	1.48E+13	12.65
WD2_rain	0.80	0.80	1.60	2.07	1.86E+13	12.95
WD2_all	0.97	0.97	0.56	0.76	1.64E+13	7.52
WD2_selection	0.93	0.93	0.83	1.20	1.06E+13	11.77
WD4_hist	0.51	0.41	6.96	11.52	5.23E+12	67.35
WD4_1lag	0.97	0.97	1.87	2.61	4.77E+12	23.79
WD4_3lags	0.98	0.98	1.73	2.36	5.42E+12	16.33
WD4_antrhop	0.95	0.95	2.43	3.51	7.77E+12	25.33
WD4_neighb	0.97	0.97	1.87	2.53	6.20E+12	20.55
WD4_meteo	0.92	0.92	3.20	4.21	9.07E+12	24.69
WD4_rain	0.85	0.84	4.71	5.93	1.02E+13	23.37
WD4_all	0.98	0.98	1.47	1.97	4.99E+12	14.71
WD4_selection	0.97	0.97	1.95	2.67	5.73E+12	22.17
WD5_hist	0.32	0.32	103.07	132.96	_	546.94
WD5_1lag	0.39	0.39	96.66	125.74	_	664.24
WD5_3lags	0.43	0.43	94.49	122.35	_	605.94
WD5_antrhop	0.42	0.42	94.77	122.51	_	720.06
WD5_meteo	0.33	0.33	102.87	132.40	_	750.52
WD5_rain	0.32	0.32	103.25	132.98	_	619.57
WD5_all	0.45	0.98	92.18	119.28	_	581.18
WD5_selection	0.43	0.98	93.83	121.57	_	605.81
WD16_hist	0.74	0.74	0.66	0.87	5.61E+10	6.87
WD16_1lag	0.84	0.84	0.50	0.69	3.63E+10	7.09
WD16_3lags	0.85	0.85	0.49	0.67	2.46E+11	7.04
WD16_antrhop	0.89	0.89	0.40	0.57	1.06E+12	7.24
WD16_neighb	0.76	0.76	0.63	0.84	2.79E+12	6.88
WD16_meteo	0.75	0.75	0.64	0.85	2.48E+10	6.88
WD16_rain	0.74	0.74	0.66	0.87	2.13E+11	6.92
WD16_all	0.00	1.00	0.58	1.12	1.28E+12	7.03
WD16_selection	0.00	1.00	0.56	1.09	2.92E+12	6.83

Table E.2: Test forecast accuracy measures obtained for each data set with the distinct ANN-based models developed.

ANN model	R^2 (-)	NSE (-)	MAE (m^3/h)	RMSE (m^3/h)	MAPE (%)	maxAE (m^3/h)
WD2_hist	0.68	0.68	2.06	2.94	20.34	19.84
WD2_1lag	0.86	0.86	1.28	1.97	12.06	20.92
WD2_3lags	0.85	0.85	1.31	1.99	12.65	21.24
WD2_antrhop	0.85	0.85	1.22	2.01	10.09	21.58
WD2_neighb	0.82	0.82	1.57	2.22	15.41	20.76
WD2_meteo	0.73	0.73	1.91	2.72	17.34	19.61
WD2_rain	0.65	0.65	2.24	3.07	24.05	19.78
WD2_all	0.82	0.81	1.57	2.26	14.70	21.35
WD2_selection	0.86	0.86	1.25	1.96	10.90	20.41
WD4_hist	0.58	0.53	6.71	10.63	3.73E+12	67.35
WD4_1lag	0.91	0.91	3.56	4.88	10.62	23.61
WD4_3lags	0.94	0.94	3.10	4.15	10.26	17.22
WD4_antrhop	0.93	0.93	3.26	4.46	9.40	24.44
WD4_neighb	0.93	0.93	3.20	4.30	10.42	21.00
WD4_meteo	0.80	0.80	5.49	7.38	16.64	28.10
WD4_rain	0.70	0.70	7.00	9.01	25.53	37.94
WD4_all	0.89	0.89	4.19	5.55	13.13	19.61
WD4_selection	0.92	0.92	3.45	4.55	11.21	19.52
WD5_hist	0.31	0.31	80.42	100.77	21.94	294.32
WD5_1lag	0.36	0.36	76.58	96.94	20.33	299.18
WD5_3lags	0.38	0.37	75.85	95.89	20.12	291.03
WD5_antrhop	0.42	0.42	71.22	92.47	18.35	310.57
WD5_meteo	0.31	0.31	79.01	100.72	22.48	351.63
WD5_rain	0.30	0.30	81.56	101.34	22.13	300.37
WD5_all	0.42	0.39	73.32	94.63	19.02	341.51
WD5_selection	0.39	0.38	74.66	95.20	19.78	311.99
WD16_hist	0.64	0.64	0.80	1.13	78.64	8.35
WD16_1lag	0.74	0.74	0.66	0.97	56.40	7.34
WD16_3lags	0.74	0.73	0.67	0.98	66.85	7.67
WD16_antrhop	0.79	0.79	0.55	0.87	41.66	7.01
WD16_neighb	0.65	0.65	0.82	1.12	79.67	8.02
WD16_meteo	0.64	0.64	0.82	1.14	77.94	7.58
WD16_rain	0.65	0.65	0.80	1.11	78.82	7.79
WD16_all	3.04	0.52	0.73	1.35	205.57	5.37
WD16_selection	3.21	0.55	0.70	1.30	188.88	4.93

Table E.3: First hour validation forecast accuracy measures obtained for each data set with the distinct ANN-based models developed.

ANN model	R^2 (-)	NSE (-)	MAE (m^3/h)	RMSE (m^3/h)	MAPE (%)	maxAE (m^3/h)
WD2_hist	_	_	3.33	3.33	28.55	3.33
WD2_1lag	_	_	3.07	3.07	26.35	3.07
WD2_3lags	_	_	2.20	2.20	18.86	2.20
WD2_antrhop	_	_	3.81	3.81	32.69	3.81
WD2_neighb	_	_	5.20	5.20	44.55	5.20
WD2_meteo	_	_	4.23	4.23	36.22	4.23
WD2_rain	_	_	4.13	4.13	35.38	4.13
WD2_all	_	_	3.29	3.29	28.21	3.29
WD2_selection	_	_	2.48	2.48	21.28	2.48
WD4_hist	_	_	4.03	4.03	9.29	4.03
WD4_1lag	_	_	2.81	2.81	6.47	2.81
WD4_3lags	_	_	3.31	3.31	7.63	3.31
WD4_antrhop	_	_	1.32	1.32	3.05	1.32
WD4_neighb	_	_	2.08	2.08	4.80	2.08
WD4_meteo	_	_	5.99	5.99	13.81	5.99
WD4_rain	_	_	3.41	3.41	7.86	3.41
WD4_all	_	_	5.86	5.86	13.51	5.86
WD4_selection	_	_	4.89	4.89	11.28	4.89
WD5_hist	_	_	124.52	124.52	56.05	124.52
WD5_1lag	_	_	153.37	153.37	69.03	153.37
WD5_3lags	_	_	104.24	104.24	46.92	104.24
WD5_antrhop	_	_	181.36	181.36	81.63	181.36
WD5_meteo	_	_	165.69	165.69	74.58	165.69
WD5_rain	_	_	128.47	128.47	57.83	128.47
WD5_all	_	_	71.37	71.37	32.13	71.37
WD5_selection	_	_	107.03	107.03	48.18	107.03
WD16_hist	_	_	0.50	0.50	20.80	0.50
WD16_1lag	_	_	0.58	0.58	24.28	0.58
WD16_3lags	_	_	0.25	0.25	10.58	0.25
WD16_antrhop	_	_	0.86	0.86	35.68	0.86
WD16_neighb	_	_	0.75	0.75	31.19	0.75
WD16_meteo	_	_	0.39	0.39	16.41	0.39
WD16_rain	_	_	0.44	0.44	18.48	0.44
WD16_all	_	_	0.65	0.65	26.98	0.65
WD16_selection	_	_	0.59	0.59	24.36	0.59

Table E.4: First 24 hours validation forecast accuracy measures obtained for each data set with the distinct ANN-based models developed.

ANN model	R^2 (-)	NSE (-)	MAE (m^3/h)	RMSE (m^3/h)	MAPE (%)	maxAE (m^3/h)
WD2_hist	0.02	-0.24	4.03	4.98	70.10	9.26
WD2_1lag	0.77	0.77	1.72	2.16	16.16	4.67
WD2_3lags	0.76	0.75	1.75	2.23	16.49	5.65
WD2_antrhop	0.74	0.73	1.59	2.33	12.96	6.24
WD2_neighb	0.75	0.72	1.87	2.38	17.65	5.43
WD2_meteo	0.57	0.43	2.78	3.36	26.95	6.88
WD2_rain	0.06	-0.74	4.56	5.89	83.49	11.52
WD2_all	0.68	0.59	2.31	2.87	22.65	6.92
WD2_selection	0.68	0.66	1.62	2.58	15.04	9.48
WD4_hist	0.02	-0.14	13.11	16.11	73.72	30.24
WD4_1lag	0.88	0.87	3.91	5.49	11.33	12.96
WD4_3lags	0.60	0.58	8.81	9.81	36.22	18.65
WD4_antrhop	0.78	0.78	5.32	7.15	14.50	16.93
WD4_neighb	0.94	0.93	3.09	3.88	10.81	8.84
WD4_meteo	0.46	0.28	10.52	12.79	32.24	24.95
WD4_rain	0.01	-0.07	13.03	15.63	67.15	30.11
WD4_all	0.58	0.30	9.88	12.61	29.63	25.85
WD4_selection	0.76	0.72	6.46	7.94	19.59	16.44
WD5_hist	0.00	-0.18	77.21	104.24	24.85	288.38
WD5_1lag	0.16	-0.11	85.56	101.29	24.09	230.65
WD5_3lags	0.25	0.01	79.79	95.58	25.47	209.81
WD5_antrhop	0.25	-0.25	75.04	107.18	20.83	284.63
WD5_meteo	0.06	-0.29	76.48	109.03	26.43	309.06
WD5_rain	0.08	0.00	68.78	95.96	23.36	282.76
WD5_all	0.29	-0.49	93.82	117.34	23.03	286.21
WD5_selection	0.20	-0.05	82.56	98.20	25.94	203.43
WD16_hist	0.01	-0.24	1.41	1.77	469.12	3.10
WD16_1lag	0.01	-0.43	1.57	1.90	495.50	3.44
WD16_3lags	0.40	0.39	1.00	1.23	275.25	2.70
WD16_antrhop	0.87	0.87	0.45	0.57	38.81	1.11
WD16_neighb	0.04	0.00	1.30	1.58	401.66	2.80
WD16_meteo	0.00	-0.04	1.40	1.62	355.65	2.91
WD16_rain	0.01	-0.10	1.41	1.66	401.16	2.80
WD16_all	0.76	0.76	0.62	0.78	49.93	2.10
WD16_selection	0.92	0.91	0.38	0.47	24.12	0.87

Table E.5: All validation data forecast accuracy measures obtained for each data set with the distinct ANN-based models developed.

ANN model	R^2 (-)	NSE (-)	MAE (m^3/h)	RMSE (m^3/h)	MAPE (%)	maxAE (m^3/h)
WD2_hist	4.31E-05	-0.01	4.82	5.89	54.37	21.02
WD2_1lag	0.69	0.69	2.36	3.28	18.81	17.63
WD2_3lags	0.68	0.68	2.45	3.29	20.41	14.83
WD2_antrhop	0.47	0.45	3.19	4.33	22.69	19.90
WD2_neighb	0.51	0.48	3.14	4.21	24.26	19.37
WD2_meteo	0.20	0.02	3.97	5.80	29.15	25.33
WD2_rain	0.00	-0.04	4.82	5.96	57.72	19.70
WD2_all	0.74	0.73	2.20	3.02	17.30	14.91
WD2_selection	0.57	0.57	2.60	3.86	17.87	23.96
WD4_hist	0.02	-0.07	16.78	20.27	54.32	58.48
WD4_1lag	0.57	0.57	9.58	12.91	21.96	45.79
WD4_3lags	0.72	0.72	7.53	10.30	20.18	40.32
WD4_antrhop	0.58	0.58	9.24	12.74	19.45	42.69
WD4_neighb	0.78	0.78	7.01	9.14	17.54	32.88
WD4_meteo	0.22	-0.26	17.12	22.04	42.43	77.52
WD4_rain	0.04	-1.09	22.78	28.32	56.84	76.30
WD4_all	0.68	0.65	9.04	11.56	22.30	43.47
WD4_selection	0.59	0.57	9.75	12.81	23.27	44.41
WD5_hist	0.00	-0.02	1.68	2.08	406.48	11.21
WD5_1lag	0.00	-0.37	1.89	2.41	403.23	12.57
WD5_3lags	0.00	-2.59	2.57	3.91	611.48	16.27
WD5_antrhop	0.64	0.63	0.86	1.25	45.63	11.39
WD5_meteo	0.05	0.02	1.67	2.04	401.38	10.78
WD5_rain	0.03	0.03	1.63	2.03	374.33	11.56
WD5_all	0.00	-0.02	1.67	2.08	336.83	11.75
WD5_selection	0.46	0.46	1.06	1.52	137.11	11.92
WD16_hist	0.42	0.38	1.01	1.62	44.14	11.81
WD16_1lag	0.03	-0.42	146.10	177.14	30.39	516.93
WD16_3lags	0.06	-3.66	273.37	321.31	49.28	772.41
WD16_antrhop	0.06	-2.07	201.96	260.76	37.73	792.60
WD16_neighb	0.02	-0.41	144.21	177.00	27.12	547.55
WD16_meteo	0.01	-0.01	121.25	149.95	28.63	449.74
WD16_rain	0.03	-0.47	149.04	180.19	30.60	522.56
WD16_all	0.13	-0.23	127.94	164.93	26.45	594.61
WD16_selection	0.04	-2.38	226.28	273.80	41.79	751.25

References

Complexo Metropolitano - sabesp. (2015). http://site.sabesp.com.br/site/interna/Default.aspx?secaoId=36. (Accessed: July 2015)

Walski, T. M., Chase, D. V., & Savic, D. (2001). *Water distribution modeling*. Haestad Press.

White, F. M. (2011). *Fluid mechanics*. McGraw-Hill.