An analysis of the interface between evolutionary algorithm operators and problem features for water resources problems. A case study in water distribution network design ☆

K. McClymont, E. Keedwell*, D. Savic

College of Engineering, Mathematics and Physical Sciences, University of Exeter, Harrison Building, North Park Road, Exeter EX4 4QF, United Kingdom

ABSTRACT

Evolutionary Algorithms (EAs) have been widely employed to solve water resources problems for nearly two decades with much success. However, recent research in hyperheuristics has raised the possibility of developing optimisers that adapt to the characteristics of the problem being solved. In order to select appropriate operators for such optimisers it is necessary to first understand the interaction between operator and problem. This paper explores the concept of EA operator behaviour in real world applications through the empirical study of performance using water distribution networks (WDN) as a case study. Artificial networks are created to embody specific WDN features which are then used to evaluate the impact of network features on operator performance. The method extracts key attributes of the problem which are encapsulated in the natural features of a WDN, such as topologies and assets, on which different EA operators can be tested. The method is demonstrated using small exemplar networks designed specifically so that they isolate individual features. A set of operators are tested on these artificial networks and their behaviour characterised. This process provides a systematic and quantitative approach to establishing detailed information about an algorithm's suitability to optimise certain types of problem. The experiment is then repeated on real-world inspired networks and the results are shown to fit with the expected results.

© 2015 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

1. Introduction

Evolutionary algorithms (EAs) have been applied to a countless number of problems across a wide variety of disciplines. Their relative simplicity and ability to work well on new problems have led to them being adopted in fields as diverse as engineering, economics and robotics. As one would expect, EAs have also been applied to water resources problems with a large degree of success, for example in the fields of groundwater remediation (Piscopo et al., 2014), controlling channel bed morphology (Nicklow et al., 2003), determining the hydraulic characteristics of production wells (Jha et al., 2004) and in particular to the field of water distribution network optimisation (e.g. Savic and Walters, 1997, Bi et al., 2014). A key aspect of EAs is that they have a number of parameters to set when first considering a new problem (e.g.

population sizes, mutation and crossover rates, selection pressure etc.) and this usually means that a period of parameter tuning is necessary to deliver acceptable performance. To alleviate this research has been carried out on algorithms to adapt these parameter settings automatically, removing the need for the parameter tuning period. A natural extension to this process then is to consider whether not just the parameters, but the operators themselves might be selected in an adaptive fashion, leading to the field known as selective hyperheuristics (Burke et al., 2013). In essence, this field explores the potential for algorithms to function beyond the strict application of selection, mutation and crossover phases and aims to develop a more dynamic approach with a greater number of operators to produce better results with less human intervention. However, to develop a suitable pool of operators from which a set might be chosen we must first understand the relationship between problem characteristics and operator function. The work herein introduces a process for exploring this interaction, and provides empirical results on a range of different artificial and real-world problems using water distribution networks as an example case study.

1.1. Water distribution network design problem

Water distribution networks (WDNs) represent one of the most complex and key infrastructures in use today and are responsible for the transportation of clean drinking water from reservoirs and storage tanks to industrial and residential consumers. Failure of these networks to adequately supply the demand can cause significant problems in the day-to-day running of businesses and homes.

A standard WDN is comprised of pipes, nodes (junctions and demand points), hydraulic devices (such as pumps) and sources (tanks and reservoirs) that constitute the entire infrastructure that delivers water from the source (e.g., reservoir) to various locations where it is drawn from the network for consumption (e.g., residential housing or industrial sites). With increasing demand and tighter regulation, water companies continue to search for more optimal operations and improvements in their networks and so have in the last two decades looked towards emerging optimisation methods to help solve their problems. Real-world WDNs are complicated structures that require constant operational management, maintenance and rehabilitation. In order to satisfy consumer demand, the networks must be constructed with a good layout that connects to all points of demand and should provide the best possible hydraulic conditions and operational requirements all whilst minimising network cost. This is known as the WDN design problem.

The WDN design problem is known to be an NP combinatorial problem (Yates et al., 1984). Even for relatively small networks, the number of possible combinations of pipes is very large which makes enumeration of all the possible designs impossible. If, for example, there were six potential sizes for each pipe in a network of just thirty pipes, there would be $6^{30} = 2.21 \times 10^{23}$ possible combinations − far more than is possible to fully enumerate within a reasonable time. This basic complexity is further compounded when advanced controls such as pump scheduling and valve operations are considered in combination with the much larger models (e.g. 1000s of pipes) that are likely to be found in the real world. Finally, when the potential for independency in looped structures and the non-linearity of the hydraulic equations is included, it becomes clear the WDN design problem is difficult non-linear, multi-modal problem. It is because of this that researchers and practitioners look to more advanced meta-heuristics to optimise their WDN designs.

1.2. Optimisation of WDNs

Since the first application of optimisation methods (Blum and Roli, 2003) to the problem of water distribution network (WDN) design researchers have collectively established a large body of literature on the subject (Marchi et al., 2014). The majority of these studies are focused on the application of novel optimisation methods to this problem, novel formulations of the problem (McClymont et al., 2013), or case studies of real-world instances. In addition, these studies have often employed or proposed new meta-heuristic methods; predominantly those from Evolutionary Computation (EC) (Coello et al., 2007) and variants of Evolutionary Algorithms (EAs) (Laumanns et al., 2000). More recently, there has been a shift in focus to hybrid (Keedwell and Khu, 2005) or more adaptive methods (Afshar, 2006) for the optimisation of these problems such as multi-method search (Vrugt and Robinson, 2007; Vrugt et al., 2009; Raad et al., 2010) and selective hyper-heuristics (McClymont et al., 2013).

The work presented here investigates search operators and their interaction with features in the fitness landscape for water distribution network optimisation, the first time this has been attempted, although attempts have been made to undertake studies of a similar nature in other domains (Franchini and Galeati, 1997). In a similar study, Zecchin et al. (2012) investigated Ant Colony Optimisation Algorithms in relation to Water Distribution Network problem characteristics and also highlighted the importance of these studies. Indeed, the optimisation of WDNs by Evolutionary Algorithms (EAs) in the early 1990s (Walters and Lohbeck, 1993; Simpson et al., 1994; Savic and Walters, 1995) was the start of a wider effort to find new, more efficient and effective optimisation techniques for this difficult real-world problem.

1.3. Performance analysis

In the search for better optimisation methods, papers frequently attempt to analyse the performance of meta-heuristics by applying them to a set of large, realistic WDNs (Walters et al., 1999; Cheung et al., 2003) or other water networks, such as in Fu et al. (2008). This experimental method provides vital information on the scalability of the proposed techniques. However, when considering the use of adaptive techniques to select operators it is important to understand the impact that individual search space features have on the behaviour and suitability of a method to that type of search space. What is required is to establish a fundamental understanding of the effect of different WDN features and landscape attributes on optimisation methodologies and lay the ground work for the new approaches described later. To make confident assertions about the true behaviour, such as the explorative or exploitative search of an algorithm, quantitative analysis of the algorithms is required (Deb and Jain, 2002; Merz, 2004).

Furthermore, while these works are important in developing better techniques for solving this class of difficult and constantly evolving real-world problems (McClymont et al., 2013), there has been relatively little work conducted on the analysis of the fundamental rules of how these optimisation methods behave under different conditions in the context of the WDN problem. Consider, for example, the significant differences in the hydraulic properties of a looped network versus a dendritic network (Walters and Lohbeck, 1993) or, similarly, a gravity fed network versus a network of pumps with tank storage. These variations in structural and hydraulic properties will result in very different optimisation search space landscapes for operators to traverse. Therefore, while one optimisation method might perform well on certain types of network, it is equally likely that it will perform less well on others. It is important to understand this relationship, between optimiser and problem, in order to make well-grounded claims about any one method's suitability for solving the WDN problem, or certain variants of it. This understanding will also help to guide algorithm and operator selection for this class of problems.

1.4. Problem and operator linkage

It is clearly shown by the No-Free-Lunch theorem (Wolpert and Macready, 1997) that not all optimisers are well suited to solving all problems. Similarly, it can be said that not all operators in an optimiser are well suited to solving all problems. This statement can be generalised somewhat to say that not all optimisation operators behave in the same way and therefore are not suited to all problems. The question therefore is: to what extent is it possible to ascertain a profile detailing the behaviour of an optimiser or its operator(s) and to determine how this profile relates to specific problems and problem features?

Malan and Engelbrecht (2013) provide some insight into the concept of characterising generalised fitness landscapes and the early work by Kauffman (1989) suggests adaptation based on these variances in the landscape is feasible. Furthermore, studies such as

Moraglio and Poli (2004) show how specific operators can have definite and identifiable attributes and behaviours (which has also been commented upon in the traditional evolutionary literature (Wright, 1932)).

This work addresses some of the current challenges facing EA research and specifically EAs for hydroinformatics as outlined in Maier et al. (submitted for publication) and is primarily concerned with exploring methods for the "development of knowledge of the underlying searching behaviour of different search methodologies". The work also touches on the "development of knowledge of the fundamental characteristics of the problem being optimised at the level at which optimisation algorithms operate" a concept that is inextricably linked with that above, as noted by Maier et al. in the section "Interaction between behaviour and fitness landscape characteristics". The behaviour of an optimiser (or part thereof) is always in response to the landscape and features of the problem being explored. One does not climb a downhill and roll up a cliff-face.

The work presented below constitutes a novel approach to quantitatively analysing and comparing different Evolutionary Algorithm operators on the WDN design problem. The method extracts key generic attributes of the WDN problem which are encapsulated in the natural features of the problem, such as network topologies and variable types, on which different EA operators can be tested. The method is demonstrated using small exemplar networks designed specifically so that they isolate individual features. A set of operators are tested on these artificial networks and their behaviour characterised. The method provides results that can be used to understand what, if any, linkages exist between the performance of an operator and certain features of a WDN, for example, the presence of pumps or the existence of looping topology. Finally, the operators are tested using real-world inspired networks for which the presence of each WDN feature is assessed. The test is a means of providing a level of confidence in the accuracy of the information learned about each operator.

2. Method

This section details the method used to compare and analyse the behaviour of EA operators and their relationship to problem features, described in general terms. The method described below is applied in this study to the WDN design problem.

2.1. A common approach to comparing methods

The majority of studies in the literature approach algorithm analysis and comparison in the same way, following from early studies and their proposed methodologies (Whitley et al., 1996; Zitzler et al., 2000). Usually, a set of benchmark problems are selected; either real-world examples or manufactured mathematical constructs and the algorithms to be compared are configured to operate effectively on the selected problems. This can be achieved by control parameter tuning on a subset of the problems or by using standard parameter values. The algorithms are then applied to (i.e., optimise) each problem and the results are collected. If the algorithm is a stochastic optimisation method, such as a meta-heuristic, then it is applied multiple times (known as trial runs) to each problem in order to collect a set of results that provide statistical information about performance. The median result is usually used for comparison for sets of trial run results. This process is illustrated in Fig. 1.

The comparative run setup is common to most studies and provides a practical and easily repeatable means for collecting data about algorithm performance. Any measure can be used, from mean or best objective value to the distribution of solutions in parameter space. The algorithms can be compared using their final solutions alone or by comparing metrics over the whole optimisation process, such as their rate of convergence, if their optimisation scales (i.e., both generational) are compatible. The benefit of this setup is that the comparison is easily understood and reflects the actual optimisation process used when practically applied to real-world problems.

The methodology presented in this paper takes this experimental approach and extends it to include preliminary steps to characterise the optimisers' performance in relation to specific features.

2.2. Method for characterising optimisers

Rather than applying the optimisers directly to problems and comparing overall performance, the method proposed here first attempts to characterise which problem features, if any, are related to an algorithm or operator's performance to provide a better insight into the underlying causes of that performance.

Unlike the common approach shown in Fig. 1, this method follows a preliminary testing phase. The method is as follows: (1) select operators, (2) select problems, (3) identify problem features, (4) synthesize artificial problems, (5) test on artificial problems, (6) analyse results and determine linkages, (7) select the most appropriate operators for selected problems, (8) test on actual problems, (9) analyse results. This is conceptually shown in Fig. 2.

Selecting the operators and problems are common tasks and not described here. The process of identifying problem features is somewhat problem dependent. For example, the WDN problem examined in this study allows for easy categorisation of problem features as the different assets and topologies of the network are the driving force between the hydraulic differences and so objective function landscapes.

2.3. Defining problem features

It is common to use mathematical test problems in addition to using real-world benchmarks. These are well defined problems with known features and can be quickly evaluated to enable large scale studies. There are many test problems available in the literature, from those identified by Van Veldhuizen (Van Veldhuizen and Lamont, 1998; Van Veldhuizen, 1999) through to more complex problem suites like the BBOB test functions (Hansen et al., 2010). This study is focused on the specifics of a real-world problem and so does not employ these test problems, although the notion of breaking down the problem into elements to construct new theoretical test problems is used here.

Furthermore, as is specified in Maier et al. (submitted for publication) (and outlined above), problem features are often described in terms of fitness landscapes: which can been seen to be the general properties of the functional mapping between parameter space and objective space. These features and the methods used to detect, analyse and capitalise on them are highly important in both the specific research of hydroinformatics but also to EA research and optimisation at large. This paper takes an alternative approach, exploring the specific features associated with the construction of a WDN. This is for two reasons: (1) the features are easily identifiable and understood by the general practitioner; and (2) the physical features of the WDN define the fitness landscape and so can be used interchangeably with those generalised features. Indeed, it should be noted that the categorisation of problems can be done on any set of features (including general ones) and is not limited to those used in this study.

2.4. Artificial problems

Synthesis of the artificial problems is similarly problem dependent. The aim is to take the set of identified features and synthesise artificial networks that represent each of these features independently and in combination with other features. For example, the presence of pumps and valves in a network present two key features in the WDN problem. Using these two features, four artificial networks can be created: no pumps or valves; valves but no pumps; pumps but no valves; both pumps and valves. These problems provide the basis for a systematic analysis of the affinity of any method to specific problem features. I.e., an EA may be well suited to problems with pumps, while others may perform better without pumps. By separating the features and selectively recombining them, it becomes possible to identify correlations in performance.
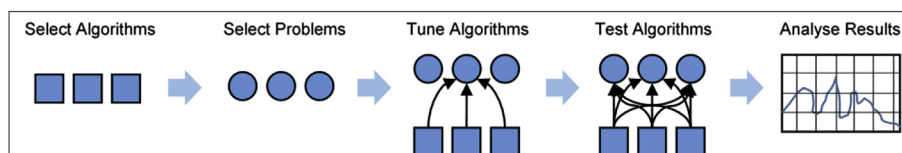

Fig. 1. A common algorithm testing process. See Fig. 3 in experimental setup for the process used in this study.
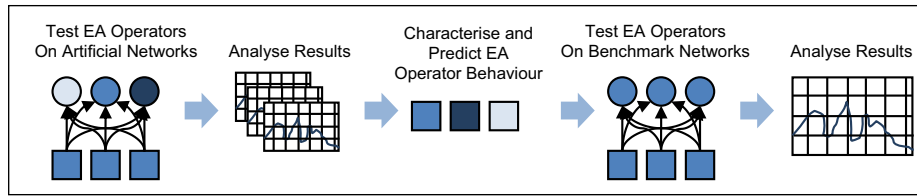
Fig. 2. Illustration of the method for characterising EA operators given different problem features encapsulated in artificial water distribution networks.

2.5. Characterising & selection of optimisers

Using the set of artificial problems, each optimiser is applied to each problem (over a number of trial runs for a fairer comparison) to produce a matrix of results. The matrix of optimisers against problems (and by proxy, features) provides a means for detecting patterns between optimisers and problem features. Given the matrix of results, it becomes possible to make a more informed selection of optimisers to apply to the larger, more expensive real-world or benchmark problems. For expensive to evaluate problems, like the WDN problem, testing large numbers of optimisers on these problems is often unfeasible given the usually limited resources. The artificial problems, by contrast, should be smaller instances of the problem and faster to execute. Indeed, in this study, the artificial problems were orders of magnitude faster in computing time to evaluate which made the wider comparison of pairs possible.

3. The water distribution network design problem

As stated above, in its basic form, a WDN is made up of pipes, junctions, demand points, hydraulic devices (such as pumps) and water sources (tanks and reservoirs). The network transports water from sources to various locations where it is drawn from the network for use. A topological illustration of an example network is given in Fig. 3.

Ideally, WDNs are constructed with a layout that connects to all demand points and provides the best possible pressure and water quality to satisfy demand. The design problem is primarily concerned with the sizing (diameters) of pipes in the network as well as scheduling of pumps and valve operations. Changing pipe sizes affects the hydraulic conditions in the network and the ability to serve the various demand points. Large pipe diameters are more expensive and so the aim of the WDN design problem is to reduce the cost of the network (minimize pipe sizes) while still satisfying customer demands (i.e., maintaining adequate pressure throughout the network). Similarly, turning pumps on and off alters the flow of the network and can increase the flow and pressure in areas of the network which are not well serviced by gravity feed alone, although the running of pumps has an additional operational cost in terms of energy consumption.

The single-objective WDN design problem is traditionally formulated as follows:

$$\text{Minimise}: \quad cost = \sum_{i=0}^{k} (c_i \times l_i) + \sum_{a=0}^{n} \sum_{b=0}^{m} \left(p_{a,b} \times e_b \right)$$
$$\text{Given}: \quad head > 30\text{m}, \text{ and } head < 40\text{m}, \text{ and } velocity < 2.5\text{ms}^{-1}$$

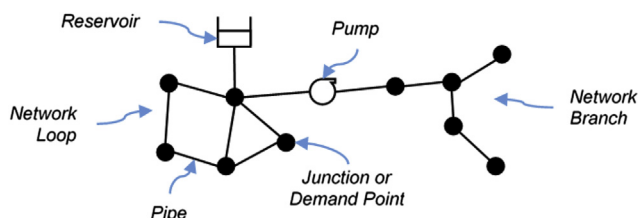This formulation aims to minimise the combined cost of the pipe infrastructure and the energy costs associated with running each pump. The symbols are as follows: $k =$ number of pipes in the network, $c_i =$ is the cost per metre of the selected diameter of pipe i (e.g. as shown in Table 1), $l_i =$ the length of pipe in metres, $n =$ the number of pumps in the network, $m =$ the number of timesteps in a simulation, $p_{a,b} =$ a binary value determining whether pump a is on at timestep b, and $e_b =$ the cost of running a pump at timestep b in US dollars. Additional constraints on the feasibility of a design are given by head pressure and velocity requirements on the network. The head and velocity constraints are considered violated if, at any time point, the network exhibits values outside of the given constraints. As noted later, the network is solved using EpaNET and these values taken from the solver's results.

3.1. WDN features

While much of the optimisation literature (beyond hydroinformatics) is concerned with the mathematical features of a problem, such as the mapping between parameters and objectives, and the landscape this creates, this study looks at the practical features present in the WDN problem. There are methods for testing for general landscape properties (multi-modality, deceptive minima, etc.) but this study looks at the base properties of the problem (topology and assets) that gives rise to these landscapes. The aim is to relate the operators to the problem, not general landscape properties. The WDN problem has a number of practical features, such as the topology of the network and the presence of different WDN assets, which directly relate to how easily the problem can be solved. The different topological features and assets are described below with a brief description of their impact on the problem.

3.1.1. Topology

There are two overarching or contrasting types of network topology: dendritic and looped networks. These extremes are described below, however it should be noted that most real-world networks are a combination of the two and are shown as 'hybrid' networks in the experiments below.

Fig. 3. An example of a water distribution network schematic (topological illustration).

Table 1
Diameter and associated costs for pipes in the artificial networks.

Diameter (mm)	Cost ($/m)
150	25
200	75
250	125
300	175
350	225
400	275
450	325
500	375
550	425
600	475
650	500

Dendritic networks have one or more water sources at the "centre" of the network. The network then extends from this source with pipes splitting into separate branches (like a tree) and serving different areas. The branches are not interconnected and so hydraulically separate to a degree. Each branch eventually terminates in an isolated end node or terminus. These networks are relatively simple to optimise as different areas have little effect on others.

Looped networks, in contrast, do not have end nodes and there are few branches in the network. Instead, the nodes are connected by "loops" that creates multiple routes of flow between the source and each demand node. These networks have more complex hydraulics and changing any one pipe can have a significant effect on many other pipes. The problem is therefore more complex with interdependencies between parameters. The networks are more robust than dendritic networks as demand nodes are not reliant on one route to the source and so can sustain a greater number of pipe failures and still function effectively. The types of loops can vary, with the most common being grid like structures that reflect the modern designs of cities.

The placement of water sources also has an effect on the network type. A network could be supplied by a single source or a concentration of sources in the same area. Equally, the sources to a network can be distributed across the network. The latter being more common when tanks are introduced in to a network.

3.1.2. Assets

Modern WDN systems are built with a wide variety of different assets each of which will have an effect on the hydraulic conditions in the network, from the essential pipes, junctions and valves to more complex units such as strainers, valves, tanks, metres and detection units. This study explores the effect of core assets which are common to most urban WDNs, which are: reservoirs, tanks, flow control valves, and time controlled pumps.

Reservoirs and tanks provide the source water to a network and dictate much of the network structure and base flow conditions of the network. Gravity fed networks (where no pumps are required to maintain adequate pressure in the network) are entirely dependent on the placement and number of water sources in the network. Reservoirs represent the most common water source in the UK whereas tanks are often used (in combination with pumps) to introduce a greater level of resilience in a network which has pressure issues and or potential for water outages from the supplying reservoir(s). Networks with more than one source are often more robust to failure and able to provide more stable pressure conditions in the network.

Valves and pumps are devices designed to manage water flow and pressure in the network to improve conditions over those given by basic gravity fed systems. Pumps allow water to be pumped upstream to areas of the network which may not be serviced by gravity fed systems or have a reduced pressure. Pumps can also be timed to accommodate increased demand at certain periods in the daily cycle. Valves are used to prevent cyclical back-flow in looped systems and control water flow, being able to restrict flow at low demand periods and opening during higher demand periods. Both of these assets make the hydraulics of a network more complex and have an effect on the pipe diameters that are optimal for the network.

The artificial networks used in this study are built from a combination of secondary reservoirs, a tank, a flow control valve, and time controlled pumps. The diameters of the pipes and the pumping schedule are used as decision variables in the search.

4. Evolutionary algorithm

An elitist Evolutionary Algorithm (EA) was used in this experiment to test different genetic operators. An EA is an iterative optimisation process which uses a population of candidate solutions which are varied to explore the search space. A standard EA takes a random initial population and then iterates through the three following processes: variation, evaluation, and selection (shown in Fig. 4). Variation is the process by which each of the existing candidate solutions are altered to create a new solution to be considered. These alterations are made using genetic operators and are most commonly mutation and/or crossover. The evaluation step assigns the objective value to each of the new candidate solutions by evaluating them on the problem (in this case the WDN problem). The selection step chooses which of the previous (parent) solutions and the new (child) solutions are kept for the next iteration to form the new set of parent solutions. This experiment varies the genetic operator in the variation step. The operators are described in a section below.

The EA used in this study used an elitist selection strategy, ranking all of the parent and child solutions and selecting the best half (minimal cost). The parents selected for crossover operators were selected at random from the elitist parent population. The EA was given a population of 20 solutions (i.e., 20 parents to produce 20 children) and was run for 1000 generations on each problem (artificial and benchmark). The EAs were trialled 100 times on each problem and the median values were taken for the results to create a more fair comparison of results and reduce the impact of outlier results in the study.

4.1. Genetic operators

As explained above, the genetic operators in an EA provide the operational process for taking one or two existing solutions and generating new candidate solutions based on these pre-existing solutions. Mutation operators take one solution, for example, and perturb the existing parameter values to create a variation on the parent solution. Crossover, on the other hand, takes two parent solutions and swaps some of the parameter values between the two parents to create two new solutions which are recombined values from both parents. The latter being an emulation of the natural mating process. In addition to these traditional operators, this study explored operators designed specifically for the WDN design problem. Each of these six operators are described below.

4.1.1. Mutation (variants)

Two mutation operators were used that mutate only one pipe diameter selected at random from the solution's parameter vector. The mutation operators were: random and 1 step size variation. The random mutation replaced the existing pipe size with a uniformly random selected pipe diameter from the set of valid pipe diameters. The 1 step mutation increased or decreased the pipe diameter (chosen at random) by one pipe size. If the pipe was the
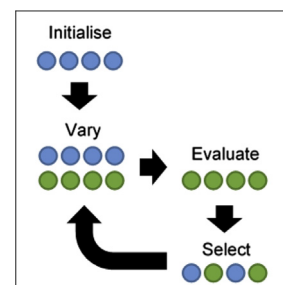

Fig. 4. Illustration of a simple Evolutionary Algorithm (where circles represent solutions and black arrows the process flow).

smallest possible size it automatically increased it. Conversely, if it was the largest possible size it decreased it.

4.1.2. Crossover (variants)

The crossover operator selected two solutions at random and recombined their values to create a new solution. There were two variants of the crossover operator: *n*-point crossover and uniform crossover.

The *n*-point crossover placed a random number of crossover points in the parameter vector and swapped the parameter values between the odd and even points to create two new solutions. The number of crossover points was between 1 and one quarter of the parameter vector length (i.e., one quarter of the number of pipes in the network). The crossover points were selected at random.

The uniform crossover operator generated a mask of random Boolean values of the same length as the parameter vector. For every parameter position that contained a true value, the parameter values were swapped. By randomly swapping parameter values, this crossover operator generated two new solutions.

4.1.3. Pipe smoothing

The locally adaptive operator (inspired by the work of Johns et al., 2013) which has a pipe smoothing effect was the first of the two specialised operators for the WDN design problem. Firstly, this operator randomly mutated a pipe size using the 1 step mutation operator. Next, the pipe smoothing operator calculated which nodes had excess head (i.e., more pressure than necessary) and placed them in a list. It then selected one of these nodes at random and decreased the upstream pipe diameter by one size, to reduce the excess head at that node. This operator therefore provided a corrective operation to the 1 step mutation operator.

4.1.4. Pipe expander

The pipe expander operator (employing an opposing operation to the pipe smoothing operator) again took the 1 step mutation operator and applied an additional pipe expansion operation. The operator worked by selecting a pipe at random and, if the downstream node had a head deficit or head less than 40 m it would increase the pipe size by one step.

5. Experimental setup

Two experiments were conducted in this study where a standard EA was used to optimise two sets of networks − one set of artificial networks and one set of benchmark networks.

In the first experiment, shown in Fig. 3, a set of different genetic operators were tested, each embedded in individual EAs. The first experiment tested each of the operators on the artificial problems to determine which features, if any, had an impact on the efficacy of the operator for optimising the single objective cost minimization problem. After each of the operators were tested, the operators were tested on the benchmark problems and the results analysed in the context of the earlier results. A second experiment was then conducted which examined the effect of pairing operators. The pairs of operators were compared on artificial problems and the impact on each of the operator's performance was analysed.

5.1. Network simulation

The hydraulics for the networks were simulated using the well-known EPANet 2.0 hydraulics solver (Rossman, 2000). All the results were calculated using the full period simulation and the worst pressure values from the whole period taken for the objective value (i.e., the lowest head for each node over the whole period).

In this experiment, 60 networks were used which varied the presence of loops, dendritic branches, combined loops and dendritic branches (hybrid), single source, double sources (close), double sources (spread), valves, and pumps. Each combination of these features were used based on the three network structures, given in Fig. 5 below. The position for the optional elements (additional sources, valves and pumps) are also shown for reference.

All the problems are constructed of 24 pipes with the same set of potential diameters and costs. The maximum total and minimum total costs for all of the artificial networks are the same, $600 to $1200 range. The diameters and associated costs per metre are given in Table 1.

For networks with pumps, the pumping costs were fixed at $50 per hour with schedules controlling the activation of pumps every 3 h allowing for a total of 8 changes to the pumping schedule in a 24 h period.

5.2. Benchmark problems

Three benchmark networks were used in the second experiment: Two Loops, Hanoi, and Anytown (Walters et al., 1999). These three benchmark networks can be downloaded from: http://emps.exeter.ac.uk/engineering/research/cws/resources/benchmarks/ and were selected as they represented well known benchmarks to compare the algorithms. Anytown presents the most complex network and is the only benchmark to include features such as pumps and tanks.

5.3. Parameter structure

The WDN problem is concerned with minimizing the cost of the network's construction with a constraint on valid values. The structure of the network, placement of sources, valves and pumps are fixed. Only the pipe diameters and, where applicable, pumping schedules could be altered in the network and as such are presented to the evolutionary algorithm as decision variables. A solution represented a list of the selected pipe size for each pipe in the network. This was represented as a vector of pipe diameter values, where the value at a given position related to a specific pipe in the network. I.e., a solution K with N pipes and 8 pump activation points is represented as $K = \{d_1, d_2, d_3, ..., d_n, p_1, p_2, ..., p_8\}$. The diameters were stored as integer numbers representing the index of the pipe size (i.e., $0 =$ smallest pipe, $1 =$ second smallest pipe, and so on). The pipe activation values were stored as integers in the set $\{0, 1\}$ where 1 indicates the pump is on for the next 3 h period and 0 indicates the pump is off for the next 3 h period. Three hour periods were selected as it reduced the size of the parameter string and fitted well with the diurnal pattern of the network.

5.4. Experimental settings

The settings for the two experiments, including evolutionary algorithm parameters are given in Table 2 below.

6. Results

6.1. General performance

Fig. 6 shows the median final objective value obtained by each of the six operators on all of the 60 artificial problems. The problems are given on the *x*-axis and the median final objective value on the *y*-axis (to be minimized). (Note: all the artificial problems objective value ranges are the same). Each of the operators are indicated with the same symbol for each objective.
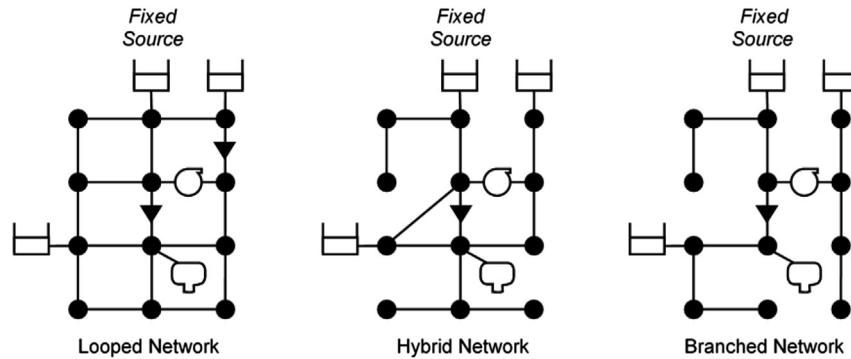
Fig. 5. The three network structures used to create the artificial networks for testing different problem features. Open boxes represent reservoir sources, circles are junctions and demand nodes, the pump is shown by a circle with an outlet, the valves as arrows and the tank as a rounded rectangle with outlet. One source is fixed while all others are optional depending on the feature(s) needed in the artificial problem.

The 1-Step Mutation operator performs the best across all the problems with the exception of some of the branched networks. This is to be expected as the mutation operator does not employ any problem specific mechanisms or make large perturbations in the search space. The crossover operators perform poorly when applied alone and this can be explained by the fact that they make large perturbations in the search space which makes it more difficult for them to converge on good solutions as they are less capable of "fine tuning" existing good solutions. Additionally, the crossover operator is limited to the set of potential solution parameter values randomly generated at the start of the search as those operators do not introduce new parameter values but rather rearrange the existing values which limits the areas of the search space that can be searched.

The specialist operators do not have a fixed performance across all the problems. As is noted later, they are better at solving the branched networks (first 20 problems) than the looped networks. This is to be expected, especially for the pipe smoothing operator, which attempts to minimize excessive pipe diameters. The operator is based on the principle that the further downstream from the source the pipes are, the smaller the diameter is required as more of the demand is upstream (although only rudimentarily implemented). The pipe expander is comparable, in performance terms, with the random mutation operator whereas the pipe smoothing operator is very efficient at solving the simpler branched networks and obtains the best results across those problems with fewer problem features.

The introduction of the pumping schedule (the last 10 sets of results in each block of 20 results) can be seen to have a visible effect on the ability of the optimisers to converge on a good result. The pipe smoothing operator is visibly affected by this on the branched networks.

Table 2
Experimental settings.

Setting	Experiment 1	Experiment 2
Number of Operators/Combinations	6	15
Artificial Networks	60	60
Trials per network	100	100
Generations	1000	1000
Population Size	20	20
Selection strategy	Elitist (truncate)	Elitist (truncate)
Archive Size	1 (best found)	1 (best found)

6.2. Convergence rates

To examine the convergence rates for each of the operators, Fig. 7 shows the number of generations taken for each operator to obtain their best objective value (averaged over all trials). As in Fig. 6, the *x*-axis displays each problem. The median number of generations to converge is given on the *y*-axis.

The standard mutation and 1-pt crossover operators converge at a similar rate across all the problems, although mutation operators converge more slowly than the crossover operator in general. The fast converging, relatively poorer results produced by the crossover operator suggests it converges early and is not able to effectively move about the search space. The mutation operators, however, are more explorative and hence converge more slowly and ultimately find better solutions than the crossover operator as is expected.

As expected, the pipe smoother converges quickly across the set of problems. Although, interestingly, it takes longer to converge on the branched networks. This is due to the continual fine tuning of the branched network to obtain the best possible network. The faster convergence on the hybrid and looped networks coupled with the worsened performance suggests the pipe smoother converges early on a poor result and is unable move away from the false optima it locates.

6.3. Loops and branches

As can be seen in Fig. 6, the predominant effect of introducing loops is to increase the final costs of the solutions discovered by all operators. In terms of relative performance among operators, the 1-step mutation remains fixed as the best performer and random mutation second across all problems. This indicates that the standard mutation operators are both robust methods for solving any WDN structure, which is expected as the operators do not employ any domain specific knowledge and only apply small perturbations to existing solutions. The presence of loops in the network did not affect the relative performance of crossover operators with respect to mutation. After examining the results produced by the crossover operator it is clear that the looped networks are more robust to large changes (due to the natural hydraulic robustness of the looped structure). However, as can be seen in Fig. 7, the looped networks are more slowly optimised by all the mutation and specialist operators (the first 20 problems).

As expected, the two specialist operators are more effective at solving the branched networks than those with loops and show a clear deterioration in performance when loops are introduced. However, the specialist operators are marginally more effective for
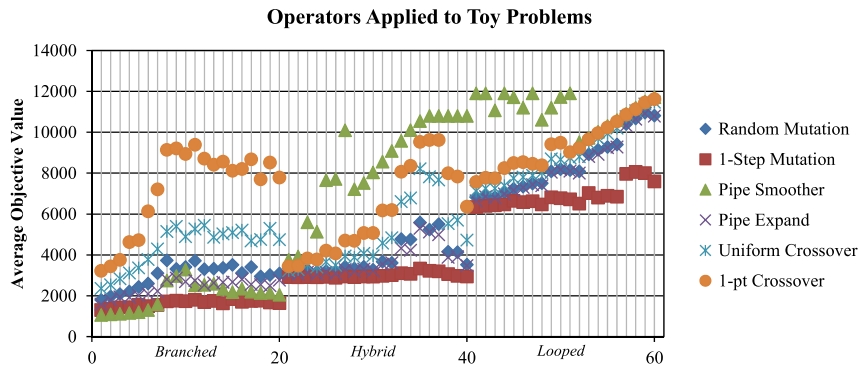
Fig. 6. Plot of the median objective value obtained by each operator on each artificial problem. The x-axis relates to each problem and the y-axis the median objective value (averaged over all the trial runs). The first 20 problems are the branched networks, the second 20 the hybrid network and the last 20 the looped networks. In each set of 20, the first 10 networks do not use pumping, in contrast to the second 10, which do.
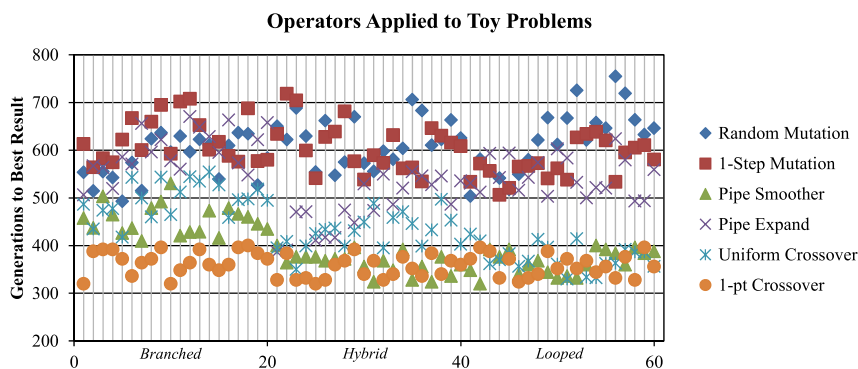

Fig. 7. A plot showing the median number of generations for each operator to find their best objective value result (converge).

solving the branched artificial problems than the standard mutation operators.

6.4. Multiple sources, pumps and valves

The presence of multiple sources does not seem to affect any of the operators in terms of convergence rates. Pumps, in contrast, significantly affect the final results for the majority of operators, particularly on the looped networks where most of the highest objective values are seen. This is not the case for the pipe smoothing operator though which shows worse performance on pumped hybrid networks, but slightly better performance on pumped looped networks. For hybrid and branched networks this can be explained as pumps interrupt the normal hydraulic conditions found in gravity fed systems and so disrupt the assumptive basis of the pipe smoothing operator (pipes further downstream generally need to supply less demand and so can be smaller).

The presence of valves (first four networks in each set of 20 results) improves the performance of the two specialist operators

as it introduces flow restrictions on the looped networks which simplifies the network hydraulics. Furthermore, the presence of pumps in the looped network improves the performance of the pipe smoothing operator, as shown in Table 3. Here we can see that generally, the introduction of pumps reduces the performance of all the operators on all the networks with the exception of the pipe smoothing operator which receives a small improvement when pumps are present in the looped network. In effect, the pump provides a similar effect to the valves in these looped networks, restricting flow. The addition of the pumps increases the complexity of the problem and the length of the chromosome. Adding pumps introduces additional parameters to be optimised and makes the search space more complex. As such, the reduced performance of the operators within a fixed computational budget is to be expected.

6.5. Combinations of operators

After examining each operator in isolation, tests were conducted to explore the effect of combining pairs of operators. Table 4 below shows the results from these tests. Each combination of operators were tested on each artificial problem for 100 trial optimisation runs. The final population median objective value from each trial was compared with the median result obtained by each operator in isolation in the previous experiment. Table 4 shows the count of trial runs that obtained better results than the median from the previous experiment.

The better than average results for operator is given in each row. For example, the row for Random Mutation shows the number of better than average results obtained by that operator when applied

Table 3
The averaged differential values between final objective function values depending on the presence of pumps in the network. Values below 1 indicate an overall reduction in performance when pumps are present in the network. Values greater than 1 indicate an improvement in performance.

Pump differential	Random mutation	1-Step mutation	Pipe smoother	Pipe expand	Uniform crossover	1-pt Crossover
Branched	0.81	0.89	0.76	0.83	0.75	0.72
Hybrid	0.72	0.94	0.65	0.74	0.58	0.53
Looped	0.77	0.91	1.07	0.77	0.77	0.81

Table 4

Frequency of improved results from combining operators over the sets of branched, hybrid and looped networks. Improvement relates to the operator given in each row when applied in conjunction with the operator assigned to each column. For example, the first row (Random Mutation) had improved results in 1815 trial runs out of 2000 when applied with the second row (1-Step mutation).

	Random Mutation	1-Step Mutation	Pipe Smoother	Pipe Expand	Uniform Crossover	1-pt Crossover
All Networks						
Random Mutation		62.15%	99.95%	55.53%	46.72%	41.77%
1-Step Mutation	30.25%		99.87%	36.00%	24.32%	19.07%
Pipe Smoother	95.83%	99.02%		97.52%	91.83%	87.77%
Pipe Expand	51.03%	62.32%	99.95%		46.10%	41.03%
Uniform Crossover	60.47%	68.95%	99.92%	62.92%		50.65%
1-pt Crossover	67.03%	76.47%	99.98%	70.18%	62.08%	
Branched Network						
Random Mutation		11.65%	99.90%	5.20%	0.80%	0.10%
1-Step Mutation	0.35%		99.65%	1.65%	0.10%	0.00%
Pipe Smoother	87.50%	97.05%		92.55%	75.50%	63.30%
Pipe Expand	1.55%	10.75%	99.90%		0.95%	0.15%
Uniform Crossover	5.60%	18.10%	99.75%	10.20%		0.95%
1-pt Crossover	14.10%	34.90%	99.95%	19.85%	6.55%	
Hybrid Network						
Random Mutation		76.85%	99.95%	65.65%	48.90%	38.80%
1-Step Mutation	19.80%		100.00%	27.90%	13.00%	7.15%
Pipe Smoother	100.00%	100.00%		100.00%	100.00%	100.00%
Pipe Expand	57.90%	77.95%	100.00%		47.85%	38.25%
Uniform Crossover	77.95%	89.70%	100.00%	80.65%		58.65%
1-pt Crossover	87.35%	94.70%	100.00%	91.20%	81.65%	
Looped Network						
Random Mutation		97.95%	100.00%	95.75%	90.45%	86.40%
1-Step Mutation	70.60%		99.95%	78.45%	59.85%	50.05%
Pipe Smoother	100.00%	100.00%		100.00%	100.00%	100.00%
Pipe Expand	93.65%	98.25%	99.95%		89.50%	84.70%
Uniform Crossover	97.85%	99.05%	100.00%	97.90%		92.35%
1-pt Crossover	99.65%	99.80%	100.00%	99.50%	98.05%	

in conjunction with each operator assigned to each column. I.e., Random Mutation and 1-Step Mutation together obtained 62.15% better than average Random Mutation alone when applied on all networks. From this it can be observed that, across the range of problems, Random Mutation performed better on average when paired with 1-Step mutation.

The results for all networks show the better than average results from a total of 6000 trial runs (accumulated across all the problems). The results for each of the three types of networks show the percentage of better than average results over an accumulated 2000 trial runs from all the networks of that type. A result of over 50% indicates the operator performs better when paired with the counter-part operator. Less than 50% indicates the operator is hindered by the other operator.

Based on the results in the previous experiment, we can see that all the operators were generally improved when applied in conjunction with 1-Step Mutation and the Pipe Smoother operator. Interestingly, 1-Step mutation was not generally improved by the addition of any other operator; with the exception of the Pipe Smoothing operator. This is an interesting result as it shows that the 1-Step Mutation was not (across the whole set of problems) improved by the addition of a crossover operator as might be expected. In contrast, the crossover operators benefitted from the addition of every other type of operator; especially the 1-pt Crossover.

The Pipe Smoothing operator is a particularly interesting case as the results highlight the "tuning" behaviour of the operator. The random perturbations created by the other operators are then "fixed" or "enhanced" by the Pipe Smoother and so it boosts the performance of these more stochastic operators. Likewise, the Pipe Smoothing operator greatly benefits from random disruptions to its early converging search behaviour and is able to better search the solution space.

When these results are examined more closely it is clear that all the stochastic operators perform better on the branched networks when applied in isolation whereas all the operators were improved by pairing when applied to the looped networks. The branched network has weaker parameter interdependencies and is therefore a simpler search space and set of problems to solve. The looped network is more difficult to solve as the loops create stronger hydraulic interdependencies between the pipes and so presents a more difficult problem. The 1-Step mutation is the only operator that has the least enhancement when applied with every operator. For example, it is not significantly improved by the addition of the 1-pt Crossover. Indeed, on the hybrid network, the 1-pt Mutation is disadvantaged by the addition of every other operator (with the exception of Pipe Smoothing). These results indicate that the generally better performance of the 1-Step Mutation is maintained, even when compared with pairings of other operators.

It is interesting to note that the 1-pt Crossover operator performs poorly across all the problems and, as shown in Table 4, provides the least improvement when applied in conjunction with other operators. This suggests that this operator is not well suited to this class of problem. Indeed, the 1-pt crossover operator is dependent upon the length and encoding type used by the problem and its application may not be well suited to the structure of this problem; as indicated by the results. That is not to say it is a bad operator, rather that it is not well suited to the problems examined here.

Tables 5 and 6 below show the median result obtained by each combination of operators on the Branched_1 and Looped_1 networks. These networks represent the 1st network of the branched and looped topologies with gravity fed systems, without pumps or tanks. In both cases the 1-Step Mutation with Pipe Smoother operator obtains the best average value compared to the other pairs. These two example sets of results are indicative of the majority of results found across all the problems.

The results show in Tables 5 and 6 demonstrate how different operators can be combined to beneficial (and detrimental) effect. Additionally, the combination of the random mutations provided by the 1-Step Mutation with the converging and problem specific action of the Pipe Smoother is a clear example of how two behaviours can be applied together in a complementary way to improve the search.

6.6. Benchmark problems

This section examines the results from the second experiment on the benchmark networks. Fig. 8 shows the median objective

Table 5

Median objective value obtained by each pairing of operator on the Branched_1 network, averaged over 100 trial runs. The best objective value is highlighted in bold.

Network: Branched_1	1-Step Mutation	Pipe Smoother	Pipe Expand	Uniform Crossover	1-pt Crossover
Random Mutation	3488.24	2884.102	3788.862	3889.922	4362.794
1-Step Mutation		**2811.675**	3260.32	3516.47	3859.401
Pipe Smoother			2850.433	2983.779	3196.36
Pipe Expand				4002.19	4214.962
Uniform Crossover					4720.72
1-pt Crossover					

Table 6
Median objective value obtained by each pairing of operator on the Looped_1 network, averaged over 100 trial runs. The best objective value is highlighted in bold.

Network: Looped_1	1-Step Mutation	Pipe Smoother	Pipe Expand	Uniform Crossover	1-pt Crossover
Random Mutation	6492.854	6124.922	6574.066	6835.196	6950.607
1-Step Mutation		**6003.437**	6410.029	6554.535	6664.378
Pipe Smoother			6060.344	6206.595	6315.911
Pipe Expand				6657.225	6818.499
Uniform Crossover					7059.035

value at the final generation for the six operators on the three benchmark networks. The assumed performance (prior to the experiment) are shown below followed by an analysis of their performance.

6.6.1. Two loops

The two loops system contains loops but no pumps and tanks and is most similar to Network 14 from the artificial problems. All the operators are expected to perform reasonably well on this network, although the two specialist operators are not expected to perform better than the normal mutation operators. As can be seen by the results, all the operators perform as expected, with the two crossover operators performing better than expected. The Two Loops network is simpler than the artificial networks and so accounts for this improved performance as the search space is smaller and the disruptive effect of the crossover operator is relatively lower compared to the single point mutation operators.

6.6.2. Hanoi

The Hanoi network contains loops and no dendritic branches. The larger number of pipes suggests the highly disruptive crossover operators will perform less well compared to the mutation operators and specialist operators due to the greater number of perturbations to the solution that will occur as a result of crossover. The gravity fed nature of this network actually produces an almost branched hydraulic system, as the pipes further away from the source need to feed only the demand of those nodes downstream. Due to the single source and this effect, there is often no requirement for flow to circulate around a loop, creating the 'branched' effect. This effective branching suggests the specialist operators will perform better on this problem.

As is shown by the results in Fig. 8, the specialist operators perform the best on this problem. These operators combine the
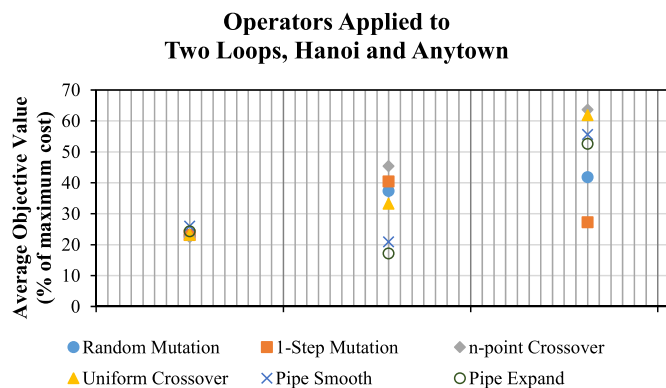

Operators Applied to Two Loops, Hanoi and Anytown

Fig. 8. The median of the final objective value obtained by each operator on the three benchmark networks. Results are shown as a percentage of the maximum cost for that network to normalise across the three networks to allow for comparison of results.

explorative power of the mutation operator and also the faster converging properties of the functions that modify the network in line with its hydraulic properties (e.g. pipe smoothing and pipe expand). The pipe smoothing operator, in particular, works very effectively on this network. As expected, the crossover operators are not able to optimise this problem in isolation. The large parameter string prevents the large perturbations of the crossover operator from converging on a solution. The mutation operators perform reasonably well again on this problem, although convergence is slower than the specialist operators as expected.

6.6.3. Anytown

This network is the most complex of the three networks tested and highly looped with the presence of additional tanks and pumps (which were scheduled as part of the optimisation). The specialist operators are not expected to perform particularly well on this problem and, similarly, the crossover operator is also not expected to perform well.

As can be seen in the results (see Fig. 8), the crossover operator is not able to effectively optimise this problem. The two specialist operators perform better than expected but converge early. The two standard mutation operators are slower to converge but produce better results by the final generation. Again, this network demonstrates how the assessment of the operators on the artificial problems enables an effective prediction of performance prior to optimising the larger, real-world or benchmark networks.

7. Conclusions

This paper has presented and given a practical demonstration of a method for assessing Evolutionary Algorithm operator behaviour in the context of specific WDN design problem features, such as pumps, tanks and loops. The method isolates the structural features of a WDN network and systematically assesses whether any correlation occurs between an operator and a feature. This method provides a means for establishing a prior understanding of an operator's efficacy on different WDN networks and aids in the development of new specialist operators and the selection of the most appropriate operators for a given network.

The paper demonstrates the method by testing six EA operators on 60 artificial networks that contain specific features. The performance of the six operators are observed and analysed and show how the more general 1-Step mutation operator performs well across the different artificial networks in comparison to the operators specialised for specific topologies (dendritic systems).

A similar comparison of the combination of operators is then conducted and the effect of combining different operators analysed. The results demonstrated how combining operators can be effective in improving EA search results in some situations. However, it is also shown that the mutation operators are more effective at solving the WDN problem on the simpler dendritic systems compared to the looped systems where combined operators are more effective. Overall, the combination of the 1-Step mutation operator and Pipe Smoothing operator is shown to be the most beneficial combination of operators.

The results presented in this paper provide evidence for the idea that operator performance and problem search spaces are linked in water distribution network optimisation, a notion that has been shown to be true in other problem spaces.

Acknowledgements

References

Afshar, M., 2006. Improving the efficiency of ant algorithms using adaptive refinement: application to storm water network design. Adv. Water Resour. 29 (9), 1371–1382.

Bi, W., Dandy, G.C., Maier, H.R. Improved genetic algorithm optimization of water distribution system design by incorporating domain knowledge. Environ. Modell. Softw. (in press), http://dx.doi.org/10.1016/j.envsoft.2014.09.010.

Blum, C., Roli, A., 2003. Metaheuristics in combinatorial optimization: overview and conceptual comparison. ACM Comput. Surv. 35 (3), 268–308.

Burke, E., Gendreau, M., Hyde, M., Kendall, G., Ochoa, G., Ozca, E., Qu, R., 2013. Hyper-heuristics: a survey of the state of the art. J. Oper. Res. Soc. 64 (12), 1695–1724.

Cheung, P.B., Reis, L.F., Formiga, K.T., Chaudhry, F.H., Ticona, W.G., 2003. Multi-objective evolutionary algorithms applied to the rehabilitation of a water distribution system: a comparative study. In: Evolutionary Multi-criterion Optimization, Lecture Notes in Computer Science, vol. 2632. Springer Verlag, pp. 662–676.

Coello, C.C., Lamont, G.B., Van Veldhuizen, D.A., 2007. Evolutionary Algorithms for Solving Multi-objective Problems, second ed. Springer, New York.

Deb, K., Jain, S., 2002. Running Performance Metrics for Evolutionary Multiobjective Optimization. Indian Institute of Technology, Kanpur.

Franchini, M., Galeati, G., 1997. Comparing several genetic algorithm schemes for the calibration of conceptual rainfall-runoff models. Hydrol. Sci. J. 42 (3), 337–379.

Fu, G., Khu, S.-T., Butler, D., 2008. Multiobjective optimisation of urban wastewater systems using ParEGO: a comparison with NSGA II. In: 11th International Conference on Urban Drainage, Edinburgh, Scotland, UK.

Hansen, N., et al., 2010. Comparing results of 31 algorithms from the black-box optimization benchmarking BBOB-2009. In: Workshop Proceedings of the GECCO Genetic and Evolutionary Computation Conference 2010. ACM.

Jha, M.K., Nanda, G., Samuel, M.P., 2004. Determining hydraulic characteristics of production wells using genetic algorithm. Water Resour. Manag. 18 (4), 353–377.

Johns, M.B., Keedwell, E., Savic, D.A., 2013. Pipe smoothing genetic algorithm for least cost water distribution network design. In: GECCO 2013, pp. 1309–1316.

Kauffman, S.A., 1989. Adaptation on rugged fitness landscapes. In: Lectures in the Sciences of Complexity. Addison-Wesley Longman, Amsterdam, pp. 527–618.

Keedwell, E., Khu, S.T., 2005. A hybrid genetic algorithm for the design of water distribution networks. Eng. Appl. Artif. Intell. 18 (4), 461–472.

Laumanns, M., Zitzler, E., Thiele, L., 2000. A unified model for multi-objective evolutionary algorithms with elitism. In: Congress on Evolutionary Computation (CEC 2000), La Jolla, CA, pp. 46–53.

Maier, H.R., Kapelan, Z., Kasprzyk, J., Kollat, J., Matott, L.S., Cunha, M.C., Dandy, G.C., Gibbs, M.S., Keedwell, E., Marchi, A., Ostfeld, A., Savic, D., Solomatine, D.P., Vrugt, J.A., Zecchin, A.C., Minsker, B.S., Barbour, E.J., Kuczera, G., Pasha, F., December 2014. Evolutionary algorithms and other metaheuristics in water resources: current status, research challenges and future directions. Environ. Model. Softw. 62, 271–299.

Malan, K.M., Engelbrecht, A.P., 2013. A survey of techniques for characterising fitness landscapes and some possible ways forward. Inf. Sci. 241, 148–163. http://dx.doi.org/10.1016/j.ins.2013.04.015.

Marchi, A., Dandy, G.C., Wilkins, A., Rohrlach, H., 2014. Methodology for comparing evolutionary algorithms for optimization of water distribution systems. J. Water Resour. Plan. Manag. 140 (1), 22–31.

McClymont, K., Keedwell, E., Savic, D., Randall-Smith, M., 2013. A general multi-objective hyper-heuristic for water distribution network design with discolouration risk. J. Hydroinform. 15 (3), 700–716. http://dx.doi.org/10.2166/hydro.2012.022.

Merz, P., 2004. Advanced fitness landscape analysis and the performance of memetic algorithms. Evol. Comput. 12 (3), 303–325.

Moraglio, A., Poli, R., 2004. Topological interpretation of crossover. In: GECCO 2004, Lecture Notes in Computer Science 3102. Springer, pp. 1377–1388.

Nicklow, J.W., Ozkurt, O., Bringer Jr., J., 2003. Control of channel bed morphology in large-scale river networks using a genetic algorithm. Water Resour. Manag. 17 (2), 113–132.

Piscopo, A.N., Kasprzyk, J.R., Neupauer, R.M., 2014. An iterative approach to multi-objective engineering design: optimization of engineered injection and extraction for enhanced groundwater remediation. Environ. Model. Softw. Available online 30 September 2014.

Raad, D., Sinske, A., Vuuren, J.v., 2010. Multiobjective optimization for water distribution system design using a hyperheuristic. J. Water Resour. Plan. Manag. 592.

Rossman, L.A., 2000. EPANET Users Manual. Technical Report No. EPA/600/R-00/057, United States 710 Environment Protection Agency.

Savic, D.A., Walters, G.A., 1995. An evolution program for optimal pressure regulation in water distribution networks. Eng. Optim. 24 (3), 197–219.

Savic, D., Walters, G., 1997. Genetic algorithms for least-cost design of water distribution networks. J. Water Resour. Plann. Manag. 123 (2), 67–77.

Simpson, A.R., Dandy, G.C., Murphy, L.J., 1994. Genetic algorithms compared to other techniques for pipe optimization. J. Water Resour. Plan. Manag. 120 (4), 423–443.

Van Veldhuizen, D.A., 1999. Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations (PhD thesis). Department of Electrical and Computer Engineering. Graduate School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, Ohio.

Van Veldhuizen, D.A., Lamont, G.B., 1998. Evolutionary computation and convergence to a pareto front. In: Koza, J.R., Banzhaf, W., Chellapilla, K., Deb, K., Dorigo, M., Fogel, D.B. (Eds.), Genetic Programming 1998: Proceedings of the Third Annual Conference. Morgan Kaufmann, pp. 221–228.

Vrugt, J.A., Robinson, B.A., 2007. Improved evolutionary optimization from genetically adaptive multimethod search. Proc. Natl. Acad. Sci. U. S. A. 708–711. http://dx.doi.org/10.1073/pnas.0610471104.

Vrugt, J.A., Robinson, B.A., Hyman, J.M., 2009. Self-adaptive Multimethod Search for Global Optimization in Real-parameter Spaces. IEEE.

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

Walters, G.A., Lohbeck, T., 1993. Optimal layout of tree networks using genetic algorithms. Eng. Optim. 22 (1), 27–48.

Whitley, D., Rana, S., Dzubera, J., Mathias, K.E., 1996. Evaluating evolutionary algorithms. Artif. Intell. 85 (1), 245–276.

Wolpert, D.H., Macready, W.G., 1997. No free lunch theorems for optimization. IEEE Trans. Evol. Comput. 1 (1), 67–82.

Wright, S., 1932. The roles of mutation, inbreeding, crossbreeding and selection in evolution. In: Jones, D. (Ed.), Proceedings of the Sixth International Congress on Genetics, pp. 356–366.

Yates, D., Templeman, A., Boffey, T., 1984. The computational complexity of the problem of determining least capital cost designs for water supply networks. Eng. Optim. 7 (2), 143–155.

Zecchin, A.C., Simpson, A.R., Maier, H.R., Marchi, A., Nixon, J.B., 2012. Improved understanding of the searching behavior of ant colony optimization algorithms applied to the water distribution design problem. Water Resour. Res. 48, W09505.

Zitzler, E., Deb, K., Thiele, L., 2000. Comparison of multiobjective evolutionary algorithms: empirical results. Evol. Comput. 8 (2), 173–195.