UNIVERSITÉ DE MONTRÉAL

MIXED-INTEGER PROGRAMMING APPROACHES FOR HYDROPOWER GENERATOR MAINTENANCE SCHEDULING

JESÚS ANDRÉS RODRÍGUEZ SARASTY
DÉPARTEMENT DE MATHÉMATIQUES ET DE GÉNIE INDUSTRIEL
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

THÈSE PRÉSENTÉE EN VUE DE L'OBTENTION
DU DIPLÔME DE PHILOSOPHIÆ DOCTOR
(MATHÉMATIQUES DE L'INGÉNIEUR)
AOÛT 2018

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Cette thèse intitulée :

MIXED-INTEGER PROGRAMMING APPROACHES FOR HYDROPOWER GENERATOR MAINTENANCE SCHEDULING

présentée par : RODRÍGUEZ SARASTY Jesús Andrés
en vue de l'obtention du diplôme de : Philosophiæ Doctor
a été dûment acceptée par le jury d'examen constitué de :

M. GENDREAU Michel, Ph. D., président
M. ANJOS Miguel F., Ph. D., membre et directeur de recherche
M. DESAULNIERS Guy, Ph. D., membre et codirecteur de recherche
M. CORDEAU Jean-François, Ph. D., membre
Mme LJUBIĆ Ivana, Ph. D., membre externe

DEDICATION

A Isaura y Kim.

ACKNOWLEDGEMENTS

First, I would like to express my gratitude to my research supervisor, Professor Miguel Anjos, for all the advice, the positive feedback, the interesting discussions, and for giving me the opportunity to develop my research interests during this PhD. I thank him for his vote of confidence and for supporting my engagement in the industrial project that motivated this dissertation.

I am also thankful to my research co-supervisor, Professor Guy Desaulniers, for his insightful comments, his thorough reviews, and his advice about the completion of my PhD. My gratitude also to Professor Charles Audet for his involvement in the initial stage of the project.

Special thanks to Dr. Pascal Côté for his invitation to work on this project, and for his participation as an industrial co-supervisor. His support was crucial for achieving the project goals. Thanks to Bruno Larouche and Jean-François Gauthier of Rio Tinto Aluminium for endorsing the financial support of the company during the 22 months of the project.

This research was also supported by grants of NSERC Engage and MITACS Accelerate programs. I also thank the Fields Institute of Mathematics of the University of Toronto, for sponsoring a poster presentation about this research at the Workshop on Nonlinear Optimization Algorithms and Industrial Applications, in Toronto, June 2016.

Thanks to the support team of GERAD (Marie, Pierre, Logo, Marilyne) and to the Énergie Électrique group of Rio Tinto in Saguenay, for providing a propitious environment for this research. Special thanks to Christophe Tribes for his assistance to set up the compilation directives for the code of this project at GERAD.

This research was motivated by an industrial problem presented at the Sixth Industrial Problem Solving Workshop, in August 2015, at the University of Montreal. I thank the organizers of that event and the Centre de Recherches Mathématiques of the University of Montreal, for creating opportunities to connect industrial problems with research in academia.

RÉSUMÉ

Dans les systèmes de production d'électricité, la maintenance régulière des unités de production est essentielle pour éviter des pannes imprévues et coûteuses, pour maintenir l'efficacité du système et pour prolonger la durée de vie de l'équipement. Cependant, l'arrêt des générateurs pour maintenance préventive réduit temporairement la capacité, l'efficacité et la fiabilité du système.

Etant donnée une liste des activités de maintenance à réaliser dans un horizon de planification, le problème de planification de maintenance des générateurs (GMSP, pour Generator Maintenance Scheduling Problem) consiste à déterminer un calendrier d'arrêts pour maintenance qui maximise une métrique de performance du système. Le calendrier optimal qui en résulte doit répondre aux exigences opérationnelles de la production d'électricité ainsi qu'aux contraintes de maintenance, telles que les fenêtres de temps des activités de maintenance.

Dans les systèmes hydroélectriques, l'ordonnancement de la maintenance des unités de production comporte des défis uniques en raison de la non-linéarité de la production d'hydroélectricité, de l'incertitude des débits d'eau et de l'interdépendance des décisions opérationnelles dans l'espace et le temps. Le GMSP est particulièrement pertinent pour les producteurs d'hydroélectricité parce que l'avancement ou le report des activités de maintenance peut générer des économies significatives en réduisant les déversements d'eau et en améliorant l'efficacité de la production d'hydroélectricité.

Nous développons un programme linéaire mixte en nombres entiers (MILP, pour Mixed-Integer Linear Program) pour le GSMP dans les systèmes hydroélectriques, avec hyperplans pour approximer l'effet non-linéaire des rejets d'eau, les niveaux d'eau stockés et le nombre de générateurs actifs sur la production d'hydroélectricité. Nous affinons notre formulation en utilisant des inégalités valides, la désagrégation de variables et de contraintes, et une technique de réduction de modèle basée sur des informations de fenêtres temporelles. Nos tests numériques montrent que la meilleure combinaison de ces techniques peut réduire jusqu'à dix fois le temps de calcul pour obtenir une solution.

Pour incorporer l'effet des afflux d'eau incertains, nous étendons notre modèle en un programme linéaire stochastique en deux étapes, et nous implémentons une méthode de décomposition de Benders parallélisée pour sa solution. Nous proposons sept techniques d'accélération, et lors de nos expériences numériques, nous observons qu'une combinaison de cinq de ces techniques permet d'obtenir les meilleures performances, avec une accélération de l'algorithme de Benders quadruplée par rapport à la méthode Benders de base. Nos tests sur une

grille de calcul avec 200 cœurs pour résoudre le problème avec un grand nombre de scénarios, confirment la supériorité de la méthode Benders parallélisée par rapport à la solution directe avec un solveur général pour MILP.

Enfin, nous proposons des extensions de notre formulation, en incluant d'autres contraintes de maintenance pertinentes, des décisions sur la durée des activités et des réserves de production pour anticiper l'incertitude de la charge d'électricité. En outre, nous présentons d'autres stratégies de décomposition pour les GMSP dans les systèmes hydroélectriques et nous discutons des perspectives de recherche, telles que des améliorations à la méthode de décomposition et les applications de notre formulation MILP à des problèmes d'ordonnancement similaires.

ABSTRACT

In power generation systems, regular maintenance of generating units is essential to prevent costly unplanned outages, to sustain the efficiency of the system, and to extend the lifespan of the equipment. However, shutting down generators for preventive maintenance temporarily reduces the capacity, efficiency, and reliability of the system.

Given a list of maintenance activities to be completed within a planning horizon, the Generator Maintenance Scheduling Problem (GMSP) is to determine a calendar of maintenance outages that maximizes a system performance metric. The resulting optimal schedule must meet operational requirements of the electricity production, as well as maintenance constraints, such as time windows of maintenance activities.

In hydropower systems, maintenance scheduling of generating units entails unique challenges due to the nonlinearity of the hydroelectricity production, the uncertainty of the water inflows and the interdependence of operational decisions in space and time. The GSMP is particularly relevant for hydropower producers because advancing or postponing maintenance activities can yield significant savings by reducing water spills and improving the efficiency of the hydroelectricity production.

We develop a compact Mixed-Integer Linear Program (MILP) for the GSMP in hydropower systems, with hyperplanes for approximating the nonlinear effect of the water discharges, the stored water levels and the number of active generators on the hydroelectricity production. We refine our formulation using valid inequalities, disaggregation of variables and constraints, and a model reduction technique based on time windows information. In computational experiments, we find that the best combination of such tightening techniques can reduce the computational time of the solution by up to one order of magnitude.

To incorporate the effect of uncertain water inflows, we extend our model as a two-stage stochastic linear program, and we implement a parallelized Benders decomposition method for its solution. We implement seven acceleration techniques, and through computational experiments, we find that a combination of five of such techniques achieves the best performance with a fourfold speedup of the Benders algorithm. Our tests on a 200-core computer cluster for solving the problem with a large number of inflow scenarios, confirm the superiority of the parallelized Benders method over the direct solution with a general MILP solver.

Finally, we outline extensions to our formulation, by including other relevant maintenance constraints, decisions on the duration of activities, and generation reserves to buffer the un-

certainty of the electricity load. Furthermore, we outline alternative decomposition strategies for the GSMP in hydropower systems and we discuss directions of future research, such as enhancements to the decomposition method and applications of our compact MILP formulation to similar scheduling problems.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

ADS	Alternative Decomposition Strategy
ANOVA	Analysis of Variance
BMP	Benders Master Problem
CBC	Combinatorial Benders cuts
CC	Combinatorial Cuts
FSP	First-Stage Problem
GCD	Greatest Common Divisor
GMSP	Generator Maintenance Scheduling Problem
HPF	Hydropower Production Function
IRC	Integer Rounding Cuts
ISO	Independent System Operator
LB	Lower Bound
LP	Linear program
MILP	Mixed-Integer Linear Program
MP	Master Problem
MPI	Message Passing Interface
MR	Multi-phase Relaxation
OP	Optimization Problem
UBF	Upper Bounding Functionals
PS	Presolve
RMP	Relaxed Master Problem
SGMSP	Stochastic (Hydropower) Generator Maintenance Scheduling Problem
SOS	Special Ordered Sets
SP	Subproblem
SSP	Second-Stage Problem
UB	Upper Bound
VI	Valid Inequalities
WS	Warm Start

CHAPTER 1 INTRODUCTION

1.1 Maintenance planning and scheduling

In a variety of systems, maintenance is an essential activity. Through effective maintenance, a system can improve its productivity, extend its life and reduce its unwanted impact on humans and the environment (Dekker, 1996). As a familiar example, a well-maintained car is not only more reliable, efficient, durable and safe, but is also less polluting.

As opposed to corrective maintenance, which occurs in response to a failure, preventive maintenance is performed to keep the system in good condition and to reduce the risk of failures. In the electricity industry, preventive maintenance reduces the risk of costly unplanned outages and can significantly increase the operating life of the system. For example, proper maintenance and rehabilitation can double the lifespan of a hydroelectric system (Fichtner, 2015).

Typical maintenance operations like inspection, cleaning, lubrication, and minor reparations involve direct costs of labour, spare parts and equipment. Less frequently, maintenance involves major investments for repairs and replacement of main components. Maintenance also entails indirect costs due to the reduction in production capacity during maintenance outages. Therefore, a cost-effective maintenance plan must determine the type and timing of maintenance activities to be performed, to balance the trade-off between the expected savings of maintenance and its overall direct and indirect costs (Dekker, 1996). However, in electric power systems, the economic impact of maintenance outages is difficult to estimate due to the uncertainty of several variables such as electricity demand, electricity prices and power generation. Maintenance planning and scheduling is further complicated by the separation of decisions in departments with conflicting objectives: whereas the maintenance department needs to perform its activities on a regular basis, the production department wants to avoid loss of production due to maintenance downtime (Budai et al., 2008).

Based on the condition of the equipment, the available resources, the production requirements and the established maintenance policies, a *maintenance plan* specifies in the mid- and long-term a list of maintenance activities to be performed, with their possible durations, required resources and time windows of execution. Given a maintenance plan, the *maintenance scheduling problem* consists in defining the execution time and sequencing of the maintenance activities to be carried out in the short-term, while respecting maintenance and production constraints (Dekker, 1996; Budai et al., 2008; Froger et al., 2016).

1.2 Challenges of maintenance scheduling in hydropower systems

Hydroelectricity is the world's main source of renewable energy, with 54.3 % of the global renewable generation capacity in 2016 (Sawin et al., 2017). Moreover, in several countries such as Norway, Canada, and New Zealand, hydropower is the main electricity source. Due to the significant role of hydropower in several territories, effective operation and maintenance of hydropower systems are essential for the reliable and efficient electricity supply. However, maintenance scheduling of generating units in hydropower systems must deal with unique challenges, such as uncertainty of water inflows, nonlinearity of the electricity production, and temporal and spatial interdependencies:

— As hydroelectricity is generated by the potential and kinetic energy of the water that drives the turbines of the system (Fig. 1.1), the total electricity production is a nonlinear function of the turbine discharges, the forebay elevation and the number of active generators (Fig. 1.2). Furthermore, generating units are also characterized by nonlinear efficiencies. All such nonlinearities add complexity to the planning and operation of hydropower systems.

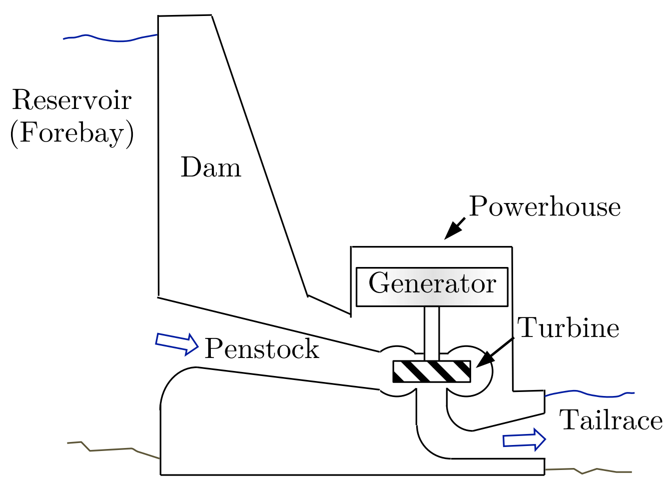


Figure 1.1 Schematic of a hydroelectric powerhouse. The stored water in the forebay flows through the penstock and propels the generating units of the system. The potential energy of the water is proportional to the net water head, which is the difference between the forebay elevation and the tailrace elevation, minus the energy losses.

— Due to the water storage capacity in hydropower systems, operating decisions are coupled in time. Immediate decisions such as water discharges determine the water levels, which impact the future operating cost of the system. Thus, maintenance activities can be postponed, anticipated or expedited to maximize the electricity production, according to the current and expected stored water levels in the system. For example, maintenance outages can be postponed when the water level is high, to reduce water spills with no economic benefit for the system.

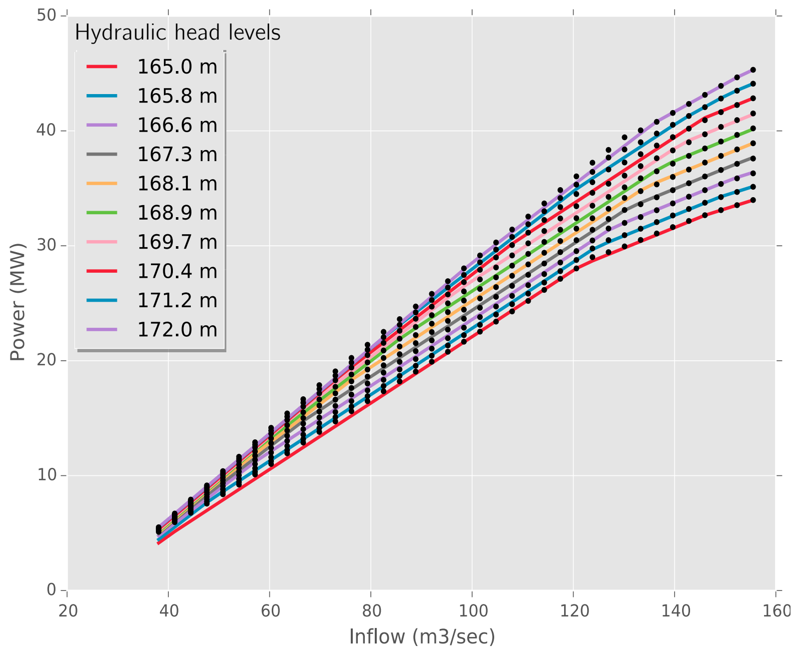


Figure 1.2 Hydroelectricity production function of a generating unit in a powerhouse. The hydraulic water head level and the water discharge have a nonlinear effect on the hydroelectricity production.

— In cascade systems, decisions are spatially interdependent: water spills and turbine discharges can feed downstream reservoirs (Fig. 1.3).

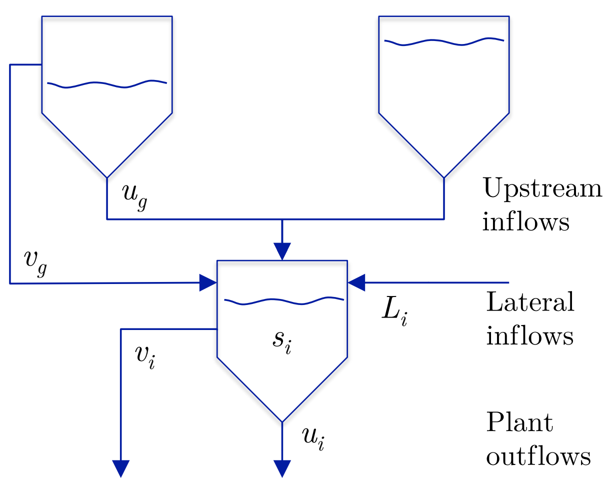


Figure 1.3 Mass balance in a reservoir. Reservoir i is fed by water discharges v and water spills u from upstream reservoirs g. Reservoir i is also fed by lateral inflows F from tributary rivers or snow-melt. Adapted from Oliveira et al. (2002).

— Reservoirs are fed by tributary rivers, snow-melt and rainfall, which can exhibit large variability and are difficult to predict (Fig. 1.4). Therefore, maintenance scheduling in hydroelectric systems must account for the uncertainty in the hydropower operation.

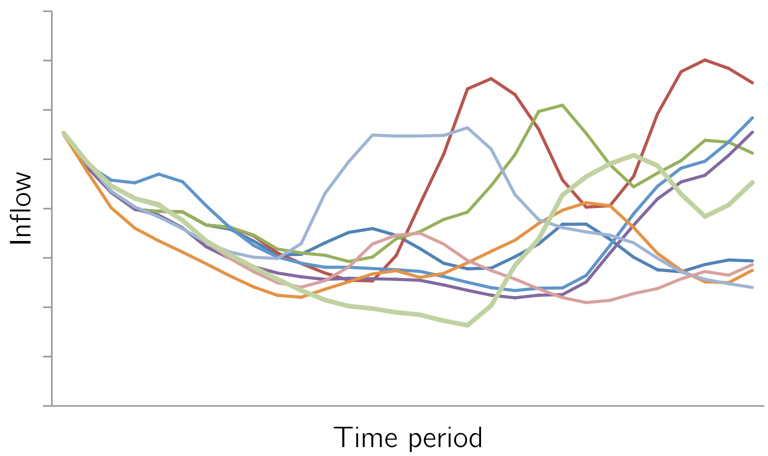


Figure 1.4 Time series of 11 forecasted water inflow scenarios

1.3 Purpose of this study

In this dissertation, we address the Stochastic Hydropower Generator Maintenance Scheduling Problem (SGMSP), i.e., the maintenance scheduling of generating units in hydropower systems, taking into account the nonlinearity of the electricity production, the uncertainty of the water inflows and the system interdependencies in space and time (Fig. 1.5). This problem can be stated as:

Given a list of maintenance activities to be completed within a specified planning horizon, find a maintenance schedule that maximizes the economic benefits of the electricity production, while considering the time windows, cost and duration of the maintenance activities, as well as maintenance constraints and essential characteristics of the hydropower operation.

This problem is motivated by a real case in Rio Tinto Aluminium, a multinational company that owns six powerhouses in Québec, with a total average generation of 2080 MW for its aluminium smelting operations. Through collaboration between two departments of the company, maintenance schedules are built manually, which can lead to delays and suboptimal schedules.

We apply mixed-integer programming techniques to obtain optimal solutions to this problem, using piece-wise linear approximations of the nonlinear hydroelectricity production. Due to the complicating aspects of the hydropower operation within maintenance scheduling, the

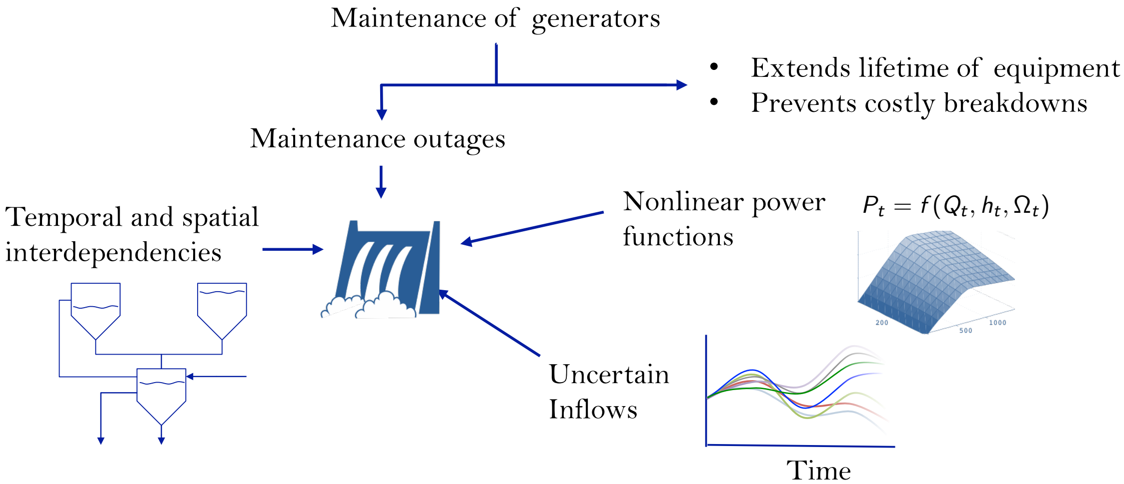


Figure 1.5 Schematic of generator maintenance scheduling in hydropower systems. Maintenance schedules impact the hydropower operation, which is also affected by uncertain water inflows, nonlinearities in the electricity production and system interdependencies.

resulting model is hard to solve for real instances of the problem. Therefore, there is a need to tighten our model and to apply Benders decomposition with parallelization and acceleration techniques to solve the SGMSP with a large number of inflow scenarios.

This work comprises the following objectives:

1. To develop a tightened mixed-integer programming formulation for the generator maintenance scheduling problem, considering the time windows of maintenance activities and the nonlinearities of the hydropower production function.
2. To implement a Benders decomposition method for the SGMSP with uncertain water inflows.
3. To accelerate the Benders decomposition method for the SGMSP by means of parallelization and acceleration techniques.
4. To propose extensions to the mathematical program for the SGMSP and to the solution approach.

1.4 Main contributions

We propose the first mixed-integer programming model for maintenance scheduling of generating units in hydropower systems, considering the time windows of maintenance activities and the nonlinear effect of turbine discharges, hydraulic head and number of active generators on the electricity production. Moreover, we extend this formulation as a two-stage stochastic program, to represent the uncertainty of the water inflows in the hydropower operation.

We develop a compact formulation for this problem (in the sense of Williams (2013)) by excluding unnecessary elements from the model. Because the generator maintenance scheduling problem is hard to solve even in the deterministic case, we tighten our model through valid inequalities, extended formulation and a set reduction method based on time windows information. Using statistical methods and computational experiments, we select the best combination of such tightening techniques based on instances adapted from a real hydropower system.

For solving the SGMSP with a large number of inflow scenarios, we implement a parallelized Benders decomposition method with seven acceleration techniques, and we show that a combination of five of these techniques achieves a fourfold speedup of the decomposition method applied to this problem. Due to a large number of potential configurations of the proposed acceleration techniques, we apply a sequential experimental design methodology to select the combination of such techniques with the best performance on the SGMSP.

We discuss extensions to our formulation for the SGMSP by including decisions on the duration of the maintenance activities and incorporating diverse maintenance constraints, such as available resources, energy reserves and precedence of activities. Finally, we show that an alternative decomposition approach, with a reduced master problem, can be applied to the SGMSP.

1.5 Plan of the dissertation

In Chapter 2, we introduce basic concepts of mixed-integer programming, Benders decomposition and maintenance scheduling, and we discuss related works. In Chapter 3 we describe the methodology of this study, and we highlight the connections between the subsequent chapters and the research objectives. Chapter 4 develops a tightened mixed-integer programming formulation for the deterministic generator maintenance scheduling problem. Chapter 5 extends this formulation as a two-stage stochastic program and develops a parallelized Benders decomposition method with acceleration techniques for its solution. In Chapter 6 we present an alternative decomposition strategy for his problem and we enhance the proposed mixed-integer formulation by considering additional maintenance scheduling decisions and requirements. A synthesis of the work, along with a discussion of the limitations and future research is presented in Chapter 7. Chapter 8 presents the main conclusions.

CHAPTER 2 LITERATURE REVIEW

2.1 Basic concepts of mixed-integer linear programming

Many real-life decision-making problems consist in selecting, with respect to a list of quantifiable criteria, the best solution among a set of alternatives. In general, these problems can be specified as mathematical optimization problems (OP) of the form

$$\underset{x}{\text{optimize}} \quad f(x), \quad x \in X \tag{OP}$$

where x is the vector of unknowns that represents the alternatives of the problem, X is the set of feasible solutions, and $f(x)$ is the function that measures the objective (or objectives when $f(x) = (f_1(x), \ldots, f_k(x))$) to be maximized or minimized, depending on the specific problem. Typical objectives in optimization problems are profit, cost, distance, travelling time, emissions and social welfare, among many others. In single-objective problems, when $f(x)$ is a linear function and the variables x are continuous with a feasible set X defined by linear inequalities, the optimization problem (OP) can be formulated as a *linear program* (LP)

$$\begin{aligned} \underset{x}{\text{maximize}} \quad & c^\intercal x \\ \text{subject to:} \quad & Ax \leq b, \\ & l \leq x \leq u, \end{aligned} \tag{LP}$$

where $A \in \mathbb{R}^{n\times m}$ is the constraint matrix, $b \in \mathbb{R}^m$ is the vector of right-hand side terms of the constraints, $c \in \mathbb{R}^n$ is the vector of the objective function coefficients, and $u, l \in \mathbb{R}^n$ are the vectors of lower bounds and upper bounds, respectively, of the decision variables $x \in \mathbb{R}^n$. Linear programs have great applicability, not only for their suitability to represent a wide variety of real problems but also because of the availability of methods for their efficient solution (Dantzig, 2002).

Due to the convexity of the polyhedron $P = \{x \in \mathbb{R}^n : Ax \leq b; \, l \leq x \leq u\}$ that describes the feasible set of LP and also because of the proportionality and additivity of the objective function $c^\intercal x$, if the feasible set P is not empty, an optimal solution to LP can always be found in one of its extreme points (Fig. 2.1). However, when some variables in x are restricted to integer values, i.e. in *mixed-integer linear programs* (MILPs), the extreme points of the *linear relaxation* solution defined by P may violate the integrality constraints (Fig. 2.2). For this reason, when solving integer linear programs, removing fractional solutions and intelligently exploring the set of integer solutions are typical approaches (Nemhauser and Wolsey, 1988).

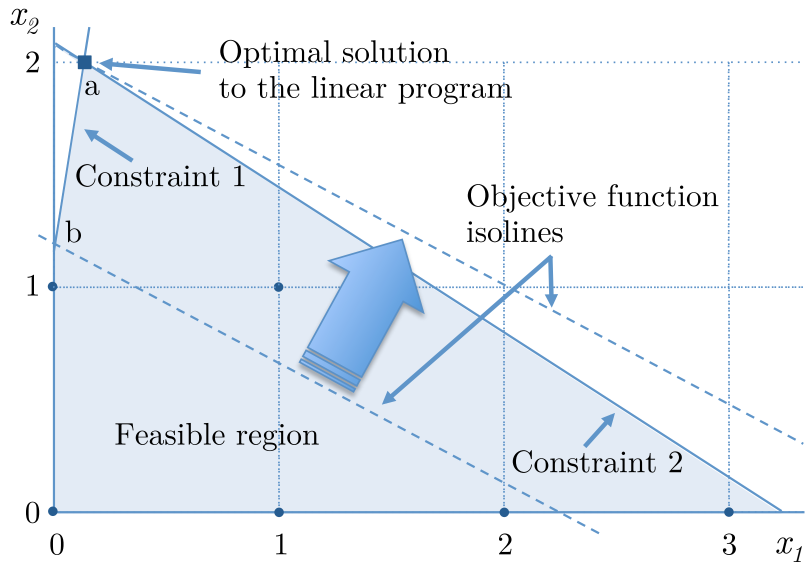


Figure 2.1 Graphical solution to a linear program with two constraints and two nonnegative variables, x_1 and x_2. The shadowed region represents the feasible set defined by constraints 1 and 2, and the variables' nonnegativity constraints. The big arrow indicates the direction of improvement of the objective function (represented by the dashed isolines). The extreme point b is feasible but not optimal. The optimal solution to the linear program is given by a, which is the last feasible point reached by the objective function in the direction of improvement.

In general, *strong* formulations lead to LP relaxation solutions with a better approximation of the MILP solution. For example, in Fig. 2.2 a linear program with a feasible region whose extreme points are integer (represented by the shadowed area) has an integer optimal solution $(3, 0)$, which is also an optimal integer solution for the original MILP. This smallest convex set that contains all integer feasible solutions is referred to as the *convex hull*.

2.1.1 General exact solution methods for mixed-integer linear programs

Because in some MILPs an exponential number of constraints would be necessary for describing the convex hull, in practice only some constraints (or cuts) are included for refining the approximation of the integer feasible set (Nemhauser and Wolsey, 1988). Such cuts can be included a priori into the model, or they can be iteratively generated through a *cutting plane* algorithm (Wolsey, 1998).

Although a complete enumeration procedure can also find the optimal solution to general

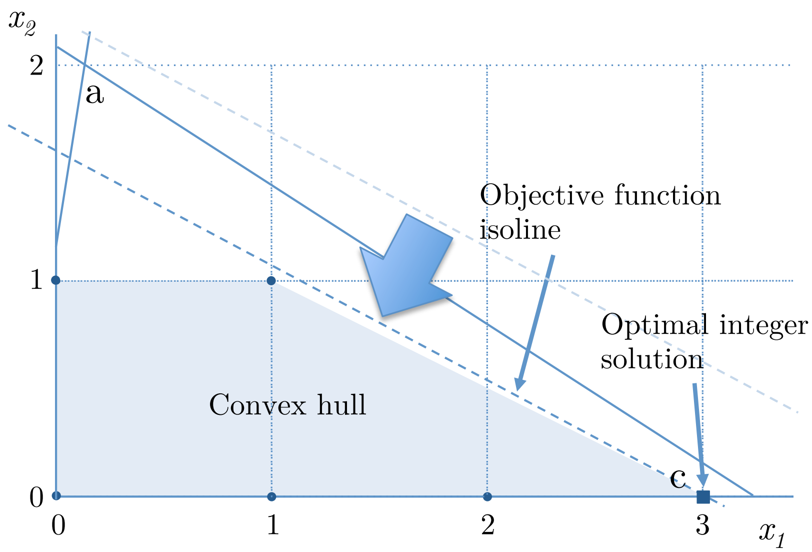


Figure 2.2 Graphical solution to the problem in Fig. 2.1 with integer variables. Since the solution to the linear program is fractional (point a), the objective function must reduce its value until reaching an optimal integer feasible solution, which occurs at the point c. Notice that the closest integer points to the fractional solution a are either infeasible or suboptimal.

.

integer programs, in general, this approach is impractical when more than a few dozen integer variables are involved, due to the combinatorial explosion of the search space (Wolsey, 1998). A more efficient strategy implicitly enumerates the solutions while avoiding the exploration of sections of the feasible region that are unlikely to contain a better solution than the *incumbent*, i.e., the current best integer solution. Such a strategy is the basic idea of the *LP-based branch and bound* method, which using an enumeration tree, systematically computes bounds based on LP relaxations and splits the search space into disjoint sets that remove fractional values of the variables, while preserving all the integer solutions of the original problem (Land and Doig, 1960).

For example, in the linear program of Fig. 2.1, at the root node of the tree the solution a has a fractional value for x_1. A branch and bound algorithm then can *branch* on this variable to create two child nodes: one with the constraint $x_1 \leq 0$, and one with $x_1 \geq 1$, which remove fractional values of x_1 and partition the feasible region as shown in Fig. 2.3. In active nodes with fractional solutions, the branching procedure continues, except when they are infeasible or when they are unpromising. For each parent node, the best LP relaxation value of its child nodes defines an upper bound. Indeed, any integer solution in either of its descendant nodes cannot be better than their LP relaxation value. Furthermore, any integer solution better than the incumbent defines a new lower bound, which is used to cut-off sections of the tree:

nodes with an LP relaxation dominated by the lower bound are fathomed (or excluded from further exploration). When the upper bound and the lower bound converge, the optimality of the incumbent solution is proven.

In practice, the execution of a branch and bound method can be time-consuming due to the slow progress of the bounds and the exponential increase of the tree size (Klotz and Newman, 2013). Cutting planes can be included at different steps of the branch and bound process, to speed up the solution by tightening up the formulation gradually. This solution approach, referred to as *branch and cut*, in combination with preprocessing, heuristics and parallel computing is an essential ingredient of state-of-the-art mixed-integer programming solvers (Ralphs et al., 2018; Bixby et al., 1999; Bixby and Rothberg, 2007).

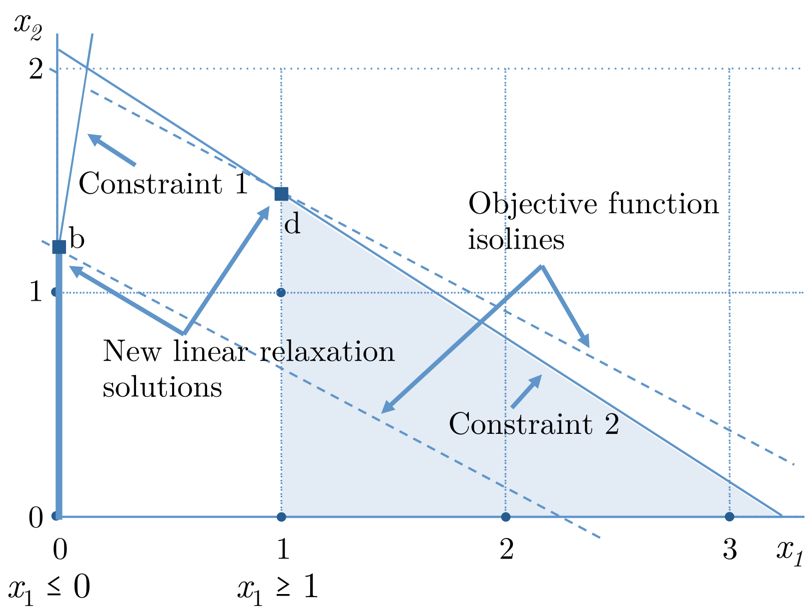


Figure 2.3 Linear relaxation solution to an integer linear program after branching on the variable x_1.

Because in hard problems the computational time to reach an optimal solution with general methods can be prohibitive, suboptimal solutions may be acceptable in practice, based on their *optimality gap*. In absolute or relative terms, the optimality gap is defined as the difference between the incumbent solution and the best bound of the relaxation.

2.1.2 Decomposition methods for mixed-integer programs

Large mathematical programs typically involve interacting subproblems linked by a few constraints or variables (Lasdon, 1970). In many cases, decomposition methods can exploit the mathematical structure associated with this class of problems referred to as *block angular*, by splitting the original model into subsystems that are solved iteratively. Other structures such as bordered angular, block triangular and staircase are also characteristic of large systems (Bradley et al., 1977).

In *primal block angular* structures the subsystems interact only through a group of linking constraints. Such constraints usually represent global conditions of the problem, such as mass balance, shared resources or total demand to be satisfied. By relaxing the linking constraints, the block angular linear program

$$
\begin{array}{llllll}
\underset{x,y}{\text{maximize}} & c_1^\intercal x_1 & + & c_2^\intercal x_2 \dots & + & c_p^\intercal x_p \\
\text{subject to:} & & & & & \\
& B_1 x_1 + & & B_2 x_2 \dots + & & B_p x_p \leq b_0 \quad \text{(Linking constraints)} \\
& A_1 x_1 & & & & \leq b_1 \\
& & & A_2 x_2 & & \leq b_2 \\
& & & \ddots & & \vdots \\
& & & & & A_p x_p \leq b_p
\end{array}
$$

splits into p independent subproblems

$$
\begin{array}{rl}
\underset{x}{\text{maximize}} & c_k^\intercal x_k \\
\text{subject to:} & A_k x_k \leq b_k,
\end{array}
$$

for $k = 1, \dots, p$.

Similarly, subsystems can be coupled by variables that usually represent decisions made in previous stages or in higher levels of the system. Such complicating variables can be subject to integrality requirements that destroy the convexity of the problem. As an example, in facility location problems, decisions on deliveries to customers are coupled across demand scenarios by strategic decisions on the capacity and location of the network's facilities.

A linear program

$$\begin{array}{lccccccccc} \text{maximize} & c_1^\intercal x_1 & + & c_2^\intercal x_2 & \dots & + & c_p^\intercal x_p & + & h^\intercal y & \\ \text{subject to:} & & & & & & & & & \\ & A_1 x_1 & & & & & & + & B_1 y & \leq b_1 \\ & & & A_2 x_2 & & & & + & B_2 y & \leq b_2 \\ & & & & \ddots & & & & \vdots & \\ & & & & & & A_p x_p & + & B_p y & \leq b_p \end{array}$$

with subsystems linked only by complicating variables y has a *dual block angular* structure. By fixing the variables $y = \bar{y}$, this linear program breaks into p independent subproblems: for $k = 1, \dots, p$,

$$\begin{aligned} \underset{x}{\text{maximize}} \quad & c_k^\intercal x_k \\ \text{subject to:} \quad & A_k x_k \leq b_k - B_k \bar{y}. \end{aligned}$$

For block angular linear programs, the two most common decomposition approaches are Lagrangian methods and methods based on the delayed generation of columns or rows of the model matrix.

Lagrangian methods relax the complicating constraints and include their violations as penalty terms in the objective function. In an iterative procedure, Lagrangian methods solve the resulting subproblems and update the penalty parameters. Such penalty parameters are approximations of the Lagrangian multipliers, which represent the prices of the shared resources or the marginal cost of the linking constraints. In problems with integer variables, Lagrangian methods are especially efficient when the underlying network structure of the subproblems can be exploited to obtain integer solutions (Ahuja et al., 1988). Some methods of this class are Lagrangian relaxation (Geoffrion, 2010), Lagrangian decomposition (Guignard and Kim, 1987) and augmented Lagrangian (Boland and Eberhard, 2015).

Another family of decomposition approaches comprises Dantzig-Wolfe decomposition (Dantzig and Wolfe, 1961) and Benders decomposition (Benders, 1962), which partition block angular mathematical programs based on the fact that a polyhedron can be described by the convex combination of its extreme solutions. Dantzig-Wolfe and Benders decomposition are duals of each other (Lasdon, 1970). Whereas Benders decomposition is suitable for dual block angular structures, Dantzig-Wolfe decomposition can be applied to primal block angular linear programs. Dantzig-Wolfe (respectively, Benders) decomposition creates a master problem where the subproblems are represented by the primal (resp. dual) contribution of their extreme solutions to the original problem. Because the set of extreme solutions of the subproblems is potentially large, the decomposition algorithm sequentially includes, as needed, the columns (resp. rows) of the master problem corresponding to the contribution of the current extreme solution (Dantzig and Wolfe, 1961; Benders, 1962). To compute the extreme solutions of the subproblems and their marginal contribution, at each iteration the decomposition algorithm fixes in each subproblem the dual (resp. primal) variables of the master problem. When the master program is integer, Dantzig-Wolfe (resp. Benders) decomposition can be embedded in a branch-and-bound method, to generate columns (resp. rows) in the nodes of the tree, when necessary (Barnhart et al., 1998; Fortz and Poss, 2009; Desaulniers et al., 1998, 2006).

Next, we give an overview of Benders decomposition, which is the solution approach applied in Chapter 5.

2.1.3 Benders decomposition

For the derivation of the Benders decomposition method (Benders, 1962), consider the mathematical program

$$\begin{aligned} \underset{x,y}{\text{maximize}}\ & c^\intercal x + f(y) \\ \text{subject to:}\ & \\ & Ax + F(y) \leq b, \\ & x \geq 0, \\ & y \in S, \end{aligned} \tag{P}$$

where y is a vector of variables with a nonconvex feasible set S that makes the whole problem hard to solve. $F(y)$ and $f(y)$ are m-component and scalar functions, respectively, and x, c, b, A are as previously defined (see LP in Section 2.1). After fixing the complicating variables $y = \bar{y}$, the resulting subproblem

$$\begin{aligned} \underset{x}{\text{maximize}}\ & c^\intercal x + f(\bar{y}) \\ \text{subject to:}\ & \\ & Ax \leq b - F(\bar{y}), \\ & x \geq 0, \end{aligned} \tag{SP}$$

is convex and thus much easier to solve. For simplicity of exposition we assume that for any $y \in \mathcal{S}$, the subproblem is feasible. By strong duality, the subproblem and its dual problem, with variables π, have the same optimal value, i.e.,

$$\begin{aligned} Q(y) &= \underset{x}{\text{maximize}}\ \{c^\intercal x : Ax \leq b - F(y)\ ;\ x \geq 0\}, \\ &= \underset{\pi}{\text{minimize}}\ \{[b - F(y)]^\intercal \pi : A^\intercal \pi \geq c\ ;\ \pi \geq 0\}. \end{aligned}$$

Furthermore, as the polyhedron $A^\intercal u \geq c\,,\ u \geq 0$ can be described by its set $\mathcal{P}$ of extreme solutions,

$$\begin{aligned} Q(y) &= \underset{p \in \mathcal{P}}{\text{minimize}}\ \{[b - F(y)]^\intercal \pi^p\}, \\ &= \underset{z^{SP}}{\text{maximize}}\ \Big\{z^{SP} \in \mathbb{R}\ :\ z^{SP} \leq [b - F(y)]^\intercal \pi^p,\ \forall p \in \mathcal{P}\Big\}, \end{aligned}$$

where the auxiliary variable z^{SP} indicates the minimum value of the dual problem. With this reformulation of the subproblem SP, the original problem P can be rewritten as

$$\begin{aligned} &\underset{z^{SP},y}{\text{maximize}}\ z^{SP} + f(y) \\ &\text{subject to:} \qquad z^{SP} \leq [b - F(y)]^\intercal \pi^p,\ \forall p \in \mathcal{P}, \\ &\qquad\qquad\qquad\quad y \in S, \end{aligned} \tag{MP}$$

which is the master problem (MP) of the Benders decomposition method. The role of constraints

$$z^{SP} \leq [b - F(y)]^\intercal \pi^p,\ \forall p \in \mathcal{P}, \tag{2.1}$$

referred to as *optimality cuts*, is to remove suboptimal solutions on the space of y, based on the extreme dual solutions of the subproblem. Due to the potentially large set of extreme solutions $\mathcal{P}$, the Benders decomposition method relaxes the set of optimality cuts (2.1), and sequentially includes violated cuts corresponding to new master problem solutions. Furthermore, in problems where a master problem solution can produce infeasible subproblems, feasibility cuts can be sequentially included, using the extreme rays of the dual subproblem. At each iteration, the upper bound is the solution value of the relaxed master problem and the lower bound is the subproblem optimal value. When the bounds converge, the optimality of the solution is proved. The Benders decomposition method can also be applied to mathematical programs with multiple subproblems, by including cuts from aggregated or disaggregated subproblem solutions (Birge and Louveaux, 1988).

In summary, the Benders decomposition algorithm solves a relaxed master problem, finds a candidate solution $\bar{y}$, solves the subproblems with the fixed candidate solution $y = \bar{y}$, verifies the optimality of the solution and checks the stopping criteria, such as computation time and number of iterations. If the solution is not optimal and the stopping criteria are not met, optimality and feasibility cuts are computed and included into the master problem, and the process is repeated (see Fig. 2.4).

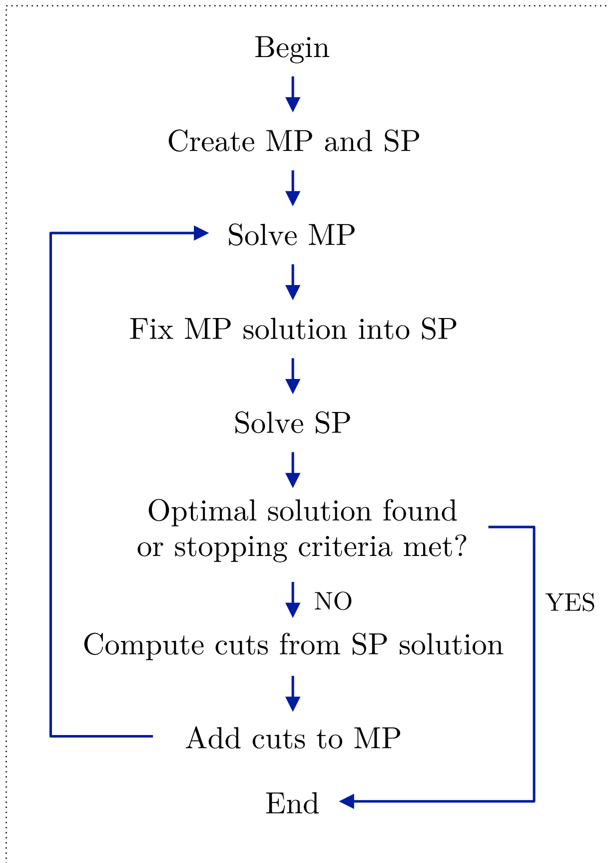


Figure 2.4 Flow diagram of the Benders decomposition method

Although Benders decomposition can, in principle, reduce the computational time by breaking a hard problem into easy-to-solve subproblems, in practice its straightforward implementation can yield poor convergence and time-consuming iterations (Rahmaniani et al., 2017; Magnanti and Wong, 1981). However, the Benders algorithm can be accelerated by adequately answering the following questions:

— **What is an ideal formulation for Benders decomposition?** Magnanti and Wong (1981) showed that, among multiple equivalent formulations of a mixed-integer program, the tightest formulation generates stronger Benders cuts and requires fewer iterations to converge. Therefore, valid inequalities and Benders cuts derived from solutions to the LP relaxation of the master problem can speed up the solution (McDaniel and Devine, 1977; Cordeau et al., 2006).

- **Which variables and constraints should be included in the master problem and the subproblems?** In non-trivial mixed-integer programs, there can be multiple ways of partitioning the problem and of including auxiliary variables and constraints that help to approximate the original problem into the master problem and the subproblems (Crainic et al., 2016; Gendron et al., 2016). The definition of the master problem and the subproblems is indeed a critical decision because it can determine the solution approach. For example, some problem partitionings can render integer subproblems or feasibility subproblems, which require special solution techniques and affect the type of cuts that can be generated (Hooker and Ottosson, 2003; Gendron et al., 2016).
- **How to efficiently obtain solutions?** Depending on the type of subproblem, special solution methods can be applied, such as network flow algorithms for network design problems (Magnanti and Wong, 1981), constraint programming for scheduling problems (Hooker and Ottosson, 2003) or Dantzig-Wolfe decomposition for crew pairing (Cordeau et al., 2001). Furthermore, when dealing with many subproblems, its parallel solution can significantly reduce the computational times. Similarly, multiple alternatives can be considered for the master problem solution, including heuristics and branch and cut (Laporte and Louveaux, 1993; Botton et al., 2013; Fortz and Poss, 2009; Fischetti et al., 2016a; Leitner et al., 2018).
- **In the case of multiple subproblem solutions, which of them should be chosen for generating cuts?** In subproblems prone to multiple dual optimal solutions, such as in network flow problems, each solution can generate a cut of different strength. Therefore, the convergence of the Benders algorithm can be accelerated by choosing a solution that generates the strongest cut (Magnanti and Wong, 1981). However, as the Magnanti and Wong (1981) approach for finding the strongest cut is computationally intensive, approximate methods have been developed for generating non-dominated cuts (Santoso et al., 2005; Papadakos, 2008; Cordeau et al., 2018).
- **What type of cuts should be generated?** When the subproblems are linear programs, optimality and feasibility Benders cuts can be generated from the extreme points and extreme rays of the dual solutions (Lasdon, 1970). In the case of integer subproblems, lower bounding functionals (Laporte and Louveaux, 1993; Carøe and Tind, 1998) or through a logic-based Benders method (Hooker and Ottosson, 2003). Recently, Angulo et al. (2016); Ljubić et al. (2017) and Álvarez-Miranda et al. (2017) showed that meaningful dual solutions of the LP-relaxation can be exploited in this context as well

for tightening the bounds of the master problem, before applying lower bounding functionals. For feasibility subproblems with binary variables, combinatorial cuts can be applied (Codato and Fischetti, 2006).

— **At each iteration, how many cuts should we generate?** In the case of multiple subproblems, at each iteration a single cut can be generated by aggregating the solutions of all subproblems. Alternatively, multiple cuts can be included by splitting the subproblems into clusters and computing a cut for each cluster at each iteration (Birge and Louveaux, 1988; Trukhanov et al., 2010). Although the multi-cut approach can reduce the number of iterations of the Benders algorithm, the effect on the computational time is problem-dependent due to the larger size of the master problem with multiple cuts (Birge and Louveaux, 2011).

— **How to improve the convergence of the decomposition algorithm?** In the Benders master problem, large step sizes in the master problem solution tend to produce oscillation and to slow down the convergence of the algorithm (Birge and Louveaux, 2011). Some stabilization approaches for Benders decomposition restrict the distance from the previous master problem solution (Santoso et al., 2005), penalize in the master problem the deviation from the previous solution (Ruszczyński and Świetanowski, 1997) or minimize the distance to a pre-defined core-point Fischetti et al. (2016a,b).

For a recent review on Benders decomposition see Rahmaniani et al. (2017), and for a survey on applications of Benders decomposition to network design problems see Costa (2005).

2.1.4 Stochastic programming

In many practical problems, decisions take place in multiple stages, with realizations of uncertain problem parameters at each stage. In these problems, decisions are made in response to the revealed information up to the current stage and to decisions made in previous stages. Therefore, optimal decisions cannot anticipate the future realizations of the problem parameters. For example, in capacity planning of electrical transmission systems, investment decisions occur in the first stage, and in subsequent stages the operating decisions of the electrical network take place in response to the electricity demand.

Although linear programming models are deterministic in nature, their application to multiple stage optimization problems under uncertainty is possible through approaches such as stochastic programming and robust optimization (Birge and Louveaux, 2011).

In stochastic linear programming, the problem uncertainty is represented by scenarios $\xi_1, \ldots, \xi_K$ with probabilities $p_1, \ldots, p_K$, which define a finitely supported joint distribution of the random problem parameters ξ. In a two-stage stochastic linear program, the first-stage problem (FSP) with decision variables y can be formulated as,

$$\begin{aligned} &\max_{y \in \mathcal{Y}} c^\intercal y + E_\xi[Q(y,\xi)] \\ &\text{subject to:} \\ &\quad Ay \leq b, \\ &\quad y \geq 0, \end{aligned} \tag{FSP}$$

where $E_\xi[Q(y,\xi)]$ is the expected optimal value of the second-stage problem (SSP)

$$\begin{aligned} &Q(y,\xi) = \max_{x \in \mathcal{X}} \ q(\xi)^\intercal x \\ &\text{subject to:} \\ &\quad T(\xi)y + W(\xi)x \leq h(\xi), \\ &\quad x \geq 0, \end{aligned} \tag{SSP}$$

with decision variables x, and random parameters T, W, h, q which depend on the specific realization of ξ. In this problem, x are the recourse actions made once the scenario parameters ξ and the decisions y are observed. The matrix W is typically assumed fixed (Shapiro et al., 2009) to characterize more conveniently the feasible region of the problem. The deterministic equivalent of the two-stage stochastic program can be obtained by replacing in FSP the expected value $E_\xi[Q(y,\xi)]$ by the weighted sum of the second-stage problems SSP, and including their corresponding constraints. The resulting mathematical program

$$\begin{array}{lllllllll} \text{maximize} & p_1 q(\xi)_1^\intercal x_1 & + & p_2 q(\xi)_2^\intercal x_2 & \ldots & + & p_K q(\xi)_K^\intercal x_K & + & c^\intercal y \\ \text{subject to:} & & & & & & & & \\ & W x_1 & & & & & & + & T(\xi)_1 y \leq h(\xi)_1 \\ & & & W x_2 & & & & + & T(\xi)_2 y \leq h(\xi)_2 \\ & & & & \ddots & & & & \vdots \\ & & & & & & W x_p & + & T(\xi)_K y \leq h(\xi)_K \\ & & & & & & & & Ay \leq b \end{array}$$

has a dual block angular structure that can be exploited via Benders decomposition when the number of scenarios is large (see Secs. 2.1.3 and 2.1.2). When the second-stage problems are integer, logic-based Benders decomposition and integer L-shaped method can be applied (Hooker and Ottosson, 2003; Laporte and Louveaux, 1993). In problems with multi-stages, common solution approaches are nested decomposition (Birge and Louveaux, 2011), progressive hedging (Rockafellar and Wets, 1991) and an extension of Benders decomposition referred to as dual dynamic programming (Pereira and Pinto, 1991).

2.2 Generator maintenance scheduling in hydropower systems

Before a revision of the previous works on maintenance scheduling of generating units in hydropower systems, we briefly discuss the principal issues in hydropower optimization and generator maintenance scheduling.

2.2.1 Planning and operation of hydropower systems

Planning and operation of hydropower systems involve decisions in multiple levels and planning horizons, ranging from several decades to a few minutes (Barros et al., 2003; Cordova et al., 2014). In the higher level, generation capacity expansion decisions are assessed according to the expected returns of the hydroelectricity production over several years, or even decades, considering the optimal operating decisions under uncertainty of water inflows and electricity demand (Gorenstin et al., 1993).

For hydropower operation, the long-term planning, which typically spans several months, computes marginal values of stored water and target levels of reservoirs based on forecasted seasonal hydrological conditions (Bezerra et al., 2017). The mid-term planning defines maintenance outages and target levels of the reservoirs, based on updated forecast information over several weeks. Short-term planning concerns the weekly operational schedule, as well as the commitment and loading of generating units. Usually, hours ahead of real-time operation, the unit commitment and loading problem determines the schedule of units for generation and ancillary services, as well as the water discharges, considering technical characteristics such as ramping rates and startup/shutdown phases of generators (Borghetti et al., 2008; Siu et al., 2001). In sub-hourly time spans, real-time control determines turbine discharges based on detailed calculations of generation efficiencies, energy losses and stored water levels (Cordova et al., 2014). Although the uncertainty of the water inflows in planning and operation of hydropower systems can be naturally addressed through stochastic dynamic programming (Bertsekas, 1995), its application in multi-reservoir systems is limited by the curse of dimensionality that results from the discretization of the stored water levels in each reservoir. Due to this challenge, a variety of alternative approaches have been applied to hydropower operation, such as dual dynamic programming (Pereira and Pinto, 1991), progressive hedging (Carpentier et al., 2013), joint chance-constrained programming (van Ackooij et al., 2014) and affine decision rules (Gauvin et al., 2017), among others.

For modelling the nonlinear hydroelectricity production, a compromise between solution quality and computational burden must be accepted. Some of the approaches for modelling the nonlinearity of such function are piece-wise linear approximations (Borghetti et al., 2008; Conejo et al., 2002; Ge et al., 2014; Marchand et al., 2018), nonlinear functions (Arce, 2001) and splines (Séguin et al., 2016) .

2.2.2 Generator maintenance scheduling

As discussed in Chapter 1, maintenance planning and scheduling are decision-making problems that consist in finding the best compromise between the opportunity costs and the expected benefits of maintenance activities. While maintenance planning defines the main necessary maintenance activities, resources and recommended times of execution, based on maintenance policies (Dekker, 1996), maintenance scheduling determines the sequence and specific times of execution of the maintenance activities within specified time windows, considering the available maintenance resources and the impact of the maintenance outages on the operational costs (Budai et al., 2008).

Due to the stochastic nature of the deterioration process and the failures, and also because of the difficulties for predicting the impact of maintenance activities on the system condition and the operation cost, the expected benefits and costs of maintenance are challenging to assess (Dekker, 1996). However, as the time windows of maintenance activities are relatively short with respect to the life-cycle of the equipment, in maintenance scheduling it is customary to neglect the equipment deterioration (Froger et al., 2016). Therefore, in maintenance scheduling the relevant costs are the opportunity cost of lost production due to maintenance outages, and the maintenance costs of overtime and outsourcing (Yamayee, 1982). All of these costs are determined by the time and duration of the activities in the maintenance schedule.

In the electricity industry, maintenance scheduling of generating units involves additional considerations:

— As a reliable electricity supply requires an instantaneous balance between consuming loads and power injections, maintenance scheduling must consider the impact of maintenance outages on the generation capacity of the system (Billinton and Allan, 1996).

— Due to the variability and uncertainty of *i*) electricity prices, *ii*) electricity demand, and *iii*) power generation from intermittent renewable energies, significant savings can be achieved by postponing, advancing or expediting the required maintenance activities.

— To accurately assess the impact of maintenance schedules, it is necessary to consider the operating characteristics of the involved electricity-generating technologies (such as eolic, nuclear and hydropower), which can differ significantly.

For vertically integrated utilities, previous works have addressed the maintenance scheduling of generating units as optimization problems with objective functions related to reliability or operational cost (Froger et al., 2016). Multi-objective approaches for resolving the conflicts between these two objectives have also been proposed (Moro and Ramos, 1999). To avoid the computational burden of probabilistic reliability measurements such as the loss of load expectation (Billinton and Allan, 1996), deterministic reliability indicators, such as generation reserve, have been commonly used in maintenance scheduling (Froger et al., 2016; Perez-Canto and Rubio-Romero, 2013).

In liberalized electricity markets, maintenance scheduling involves additional complications due to the interacting decisions of multiple self-interested market participants, such as private electricity producers, transmission companies and electricity retailers who want to maximize their profits. In such markets, an Independent System Operator (ISO) coordinates and controls the market activities to maximize the social welfare while guaranteeing the reliability of the electricity supply. Maintenance schedules proposed by the market participants must be revised and coordinated by the ISO through mechanisms such as contractual compensations (Dahal et al., 2015) or incentives (Conejo et al., 2005), to ensure the reliability of the system.

Modelling all the aforementioned elements involved in generator maintenance scheduling leads to a hard problem for which typical solution approaches are heuristics, metaheuristics and mixed-integer programming with Benders decomposition (Froger et al., 2016).

2.2.3 Maintenance scheduling of generating units in hydropower systems

Despite the vast body of literature on maintenance scheduling in the electrical industry (see Froger et al., 2016, for a recent review), few works to date have addressed this problem in the context of hydroelectric systems. Furthermore, some works on maintenance scheduling that claim to have considered hydroelectric systems, have entirely neglected or oversimplified the most relevant aspects of the hydropower operation, such as the stored water effects, the uncertainty of the water inflows and the nonlinearities of the electricity production (Feng et al., 2011; Foong et al., 2008; Perez-Canto, 2008; Perez-Canto and Rubio-Romero, 2013; Chattopadhyay et al., 1995; Kuzle et al., 2010).

For long-term planning of preventive maintenance in hydropower systems, Jonsson (2015) used a dependency matrix for determining the sequence of activities in maintenance projects, and Welte et al. (2006) developed a Markov model of the equipment deterioration. Such models did not represent the main operational aspects of the hydroelectricity production.

Chattopadhyay et al. (1995) formulated a mixed-integer program for the coordinated maintenance scheduling between interconnected utilities with different generation technologies. Their formulation did not include the operational characteristics of hydropower systems and assumed a fixed amount of available energy at each time period. For a deterministic generator maintenance scheduling problem, Foong et al. (2008) developed an ant colony approach that uses simple heuristic rules for determining the turbine discharges, without regarding the effects of the nonlinear hydroelectricity production. Feng et al. (2011) represented the variability of the electricity generation with fuzzy variables but did not consider the relevant characteristics of the hydropower operation. Perez-Canto (2008), Perez-Canto and Rubio-Romero (2013) and Kuzle et al. (2010) presented mixed-integer programming formulations for maintenance scheduling, which assume constant power output in active generators and do not represent the operating characteristics of hydropower systems. For such formulations, Perez-Canto (2008) and Kuzle et al. (2010) applied Benders decomposition and Perez-Canto and Rubio-Romero (2013) proposed to search only for feasible solutions.

More realistic elements of hydroelectric systems have been considered in Régnier (2008), Guedes et al. (2015), Côté et al. (2015), Helseth et al. (2018) and Ge et al. (2018).

Régnier (2008) developed a heuristic method that uses an auxiliary function for assessing the effect of candidate schedules on the hydropower operation. Applying a black-box approach, Côté et al. (2015) showed that important savings can be achieved by exploring the neighbourhood of an initial maintenance schedule, even if the new solution is not globally optimal. Guedes et al. (2015) implemented a genetic algorithm for a deterministic version of the generator maintenance scheduling problem, with continuous variables for representing the starting times of maintenance activities within time window constraints. Although Guedes et al. (2015) represented the hydroelectricity production with an analytic function, their model neglects the nonlinear effect of the number of maintenance outages on the amount of produced electricity. More recently, Ge et al. (2018) developed a chance-constrained approach for maintenance scheduling, with a piece-wise approximation of the hydroelectricity production, as in Ge et al. (2014), but without considering the nonlinear effect of the number of active generators. Helseth et al. (2018) introduced a different approach for modelling the uncertainty in this problem, through a multi-stage stochastic optimization program with maintenance decisions in the first-stage and hydropower operation in multiple stages, solved by dual dynamic programming. Helseth et al. (2018) represented with a scenario tree the uncertainty of water inflows and demand, and approximated with piecewise segments the non-linearity of the hydroelectricity production with respect to turbine discharges. However, their formulation did not consider the nonlinear effect of reservoir levels and of the set of active generators, which substantially reduces the problem size but cannot realistically represent the nonlinearity of the hydroelectricity production function.

Concerning mixed-integer formulations of maintenance scheduling in the literature, we found three approaches:

1. Helseth et al. (2018) defined binary variables only for indicating the state of the generators (i.e., active or in maintenance, as in Conejo et al. (2005)). Through algebraic constraints, this formulation explicitly controls the duration and continuity of maintenance activities across consecutive periods.
2. In Perez-Canto (2008) and Perez-Canto and Rubio-Romero (2013), two types of binary variables indicate the state of the generator and the beginning of maintenance activities. In such formulations, explicit constraints link the binary variables and control the duration and continuity of maintenance activities across consecutive periods.
3. Dahal et al. (2015) proposed a compact formulation, where only binary variables for indicating the beginning of maintenance activities are necessary and with the time windows controlled through index sets. Because in this approach the duration of maintenance activities is controlled by index sets and not by algebraic constraints, its corresponding formulations are thinner and stronger than those corresponding to formulation approaches 1 and 2. However, this work did not address specific issues of maintenance scheduling in hydropower systems.

Our review of the literature on maintenance scheduling of generating units in hydropower systems, supports the following conclusions:

— Although the nonlinearity of the hydroelectricity production has been considered in recent works for obtaining globally optimal solutions to the problem (Helseth et al., 2018; Ge et al., 2018), the effect of stored water levels and number of active generators on the hydroelectricity production has not yet been appropriately addressed. Neglecting such elements can lead to poor estimates of the hydroelectricity production and to suboptimal solutions in practice.

— Considering the significant variability of the water inflows and their effect in the hydropower operation, alternative representations of the uncertainty, in addition to recently proposed chance-constrained (Ge et al., 2018) and scenario tree approaches (Helseth et al., 2018), should be explored to achieve an acceptable compromise between solution quality and computational tractability.

— A more realistic representation of maintenance scheduling decisions is necessary, considering other relevant objectives and constraints.

— Stronger formulations and alternative solution methods must be explored to find efficiently global optimal solutions to real instances of the problem, instead of local optimal solutions.

Considering these gaps in the literature, in the following section, we describe our approach for developing more realistic and efficient solutions to this problem.

CHAPTER 3 THESIS ORGANIZATION

In the following chapters, we develop a mixed-integer linear program and a decomposition-based solution method for maintenance scheduling of generating units in hydropower systems (SGMSP), considering the uncertainty of the water inflows and the nonlinearities of the hydroelectricity production (Fig. 3.1).

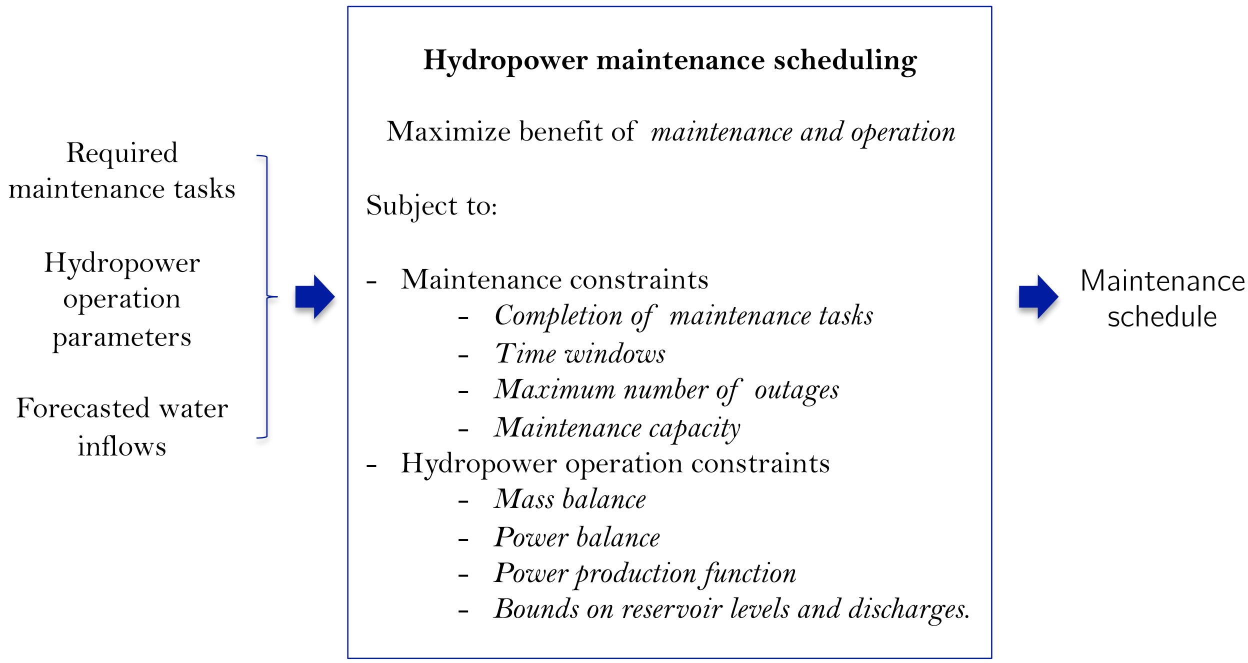


Figure 3.1 Input-output diagram of the hydropower maintenance scheduling problem

For this purpose, in Chapter 4, we develop a MILP formulation for a deterministic version of this problem. Because the resulting model has a weak continuous relaxation and is hard to solve, we improve our formulation through valid inequalities and reduction of the model size. We also develop an alternative (*extended*) formulation with disaggregated variables and constraints, which allows defining tighter bounds of the hydroelectricity production. Moreover, in Chapter 4 we run computational experiments to select the best combination of tightening techniques. In Chapter 5 we extend the tightened formulation to consider the effect of uncertain water inflows on the hydropower operation. Thus, we develop a two-stage stochastic programming model with maintenance scheduling decisions in the first stage and operational decisions with inflow scenarios in the second stage.

To accurately represent the significant uncertainty of the water inflows, several inflow scenarios should be considered in the SGMSP. As the straightforward solution of the resulting large-scale problem is hard to achieve, in Chapter 5 we implement a Benders decomposition method for splitting the SGMSP into scenario-wise operational subproblems and a master problem with maintenance scheduling decisions. We explore seven acceleration techniques for this decomposition approach and through computational experiments we determine the combination of such techniques with the lowest computational times. Also, in Chapter 5 we parallelize the Benders algorithm for the SGMSP, and we test its execution on a 200-core computer cluster.

In Chapters 4 and 5, the formulation for the SGMSP was motivated by an industrial problem in Rio Tinto Aluminium. To address more general maintenance scheduling problems, in Chapter 6 we extend the MILP formulation of Chapter 5 with additional maintenance constraints and discrete choice of duration of activities. Furthermore, we propose chance-constrained and max-min formulations to consider the uncertainty of the electricity load. In Chapter 6 we also outline an alternative decomposition strategy for the SGMSP, using a reduced master problem.

In Chapter 7 we summarize the research outcomes, and we discuss the limitations of this study and directions for future research.

Table 3.1 summarizes the connection between the research objectives, the proposed approaches and the following chapters of this dissertation.

Table 3.1 Summary of thesis organization

<table>
<tr><th>Objective</th><th>Approach/Technique</th><th>Chapter</th></tr>
<tr><td rowspan="4">1. To develop a tightened mixed-integer programming formulation for the generator maintenance scheduling problem, considering the time windows of maintenance activities and the nonlinearities of the hydropower production function</td><td>MILP formulations</td><td rowspan="3">4</td></tr>
<tr><td>Tightening techniques for MILP</td></tr>
<tr><td>Computational experiments</td></tr>
<tr><td>Stochastic programming</td><td rowspan="5">5</td></tr>
<tr><td>2. To implement a Benders decomposition method for the SGMSP with uncertain water inflows.</td><td>Benders decomposition</td></tr>
<tr><td rowspan="3">3. To accelerate the Benders decomposition method for the SGMSP by means of parallelization and acceleration techniques.</td><td>Parallel programming</td></tr>
<tr><td>Acceleration techniques for Benders decomposition</td></tr>
<tr><td>Computational experiments</td></tr>
<tr><td rowspan="3">4. To outline extensions to the mathematical program for the SGMSP and to the solution approach.</td><td>MILP formulations: additional variables and constraints</td><td rowspan="3">6</td></tr>
<tr><td>Considering reserves: chance-constrained and max-min formulations</td></tr>
<tr><td>Alternative decomposition strategy</td></tr>
</table>

CHAPTER 4 ARTICLE 1: MILP FORMULATIONS FOR GENERATOR MAINTENANCE SCHEDULING IN HYDROPOWER SYSTEMS

Authors: Jesus A. Rodríguez, Miguel F. Anjos, Pascal Côté, Guy Desaulniers
Accepted for publication: IEEE Transactions on Power Systems, May 2018.

Abstract: Maintenance activities help prevent costly generator breakdowns but because generators under maintenance are typically unavailable, the impact of maintenance schedules is significant and their cost must be accounted for when planning maintenance. In this paper we address the generator maintenance scheduling problem in hydropower systems. We propose a mixed-integer programming model that considers the time windows of the maintenance activities, as well as the nonlinearities and disjunctions of the hydroelectric production functions. Because the resulting model is hard to solve, we also propose an extended formulation, a set reduction approach that uses logical conditions for excluding unnecessary set elements from the model, and valid inequalities. We performed computational experiments using a variety of instances adapted from a real hydropower system in Canada, and the extended formulation with set reduction achieved the best results in terms of computational time and optimality gap.

Keywords: Hydroelectric power generation, Integer linear programming, Mathematical programming, Optimal maintenance scheduling

4.1 Notation

We denote decision variables and indices with lowercase, and parameters with uppercase.

Primary sets

$\mathcal{I}$: Powerhouses

$\mathcal{M}$: Maintenance tasks

$\mathcal{T}$: Planning time periods, $t \in \mathcal{T} = \{1 \dots T\}$.

Parameters

B_t^-, B_t^+ : Prices of electricity purchase and sale, respectively, at time period t, where $B_t^- \geq B_t^+$ [$/MWh].

C_{mt} : Total cost of maintenance task m started at time period t [$].

D_m : Duration of maintenance task m [day].

E_m : Earliest start time period of maintenance task m.

F_{it} : Lateral inflows to powerhouse i in period t [m^3/s].

$\bar{G}_{it}, \underline{G}_i$: Limits on the number of available turbines in powerhouse i at time period t [turbines].

J_t : Electricity load at time t [MWh].

L_m : Latest start time period of maintenance task m.

O_{it} : Maximum number of turbine outages in powerhouse i at time period t [turbines].

$\bar{P}_i$: Generation capacity in powerhouse i [MWh/day].

$\bar{P}_{ik}$: Generation capacity in powerhouse i when k turbines are active [MWh/day].

Q : Factor for conversion from m^3/s to hm^3/day [$0.0864 \cdot s \cdot hm^3 \cdot / (day \cdot m^3)$].

$\bar{R}_{it}$: Number of maintenance activities that *can* be in execution at powerhouse i in time period t.

$\underline{R}_{it}$: Number of maintenance activities that *must* be in execution at powerhouse i in time period t.

S_{0i} : Initial stored water in reservoir of powerhouse i [hm^3].

$\underline{S}_i, \bar{S}_i$: Limits on stored water in reservoir of powerhouse i [hm^3].

$\bar{U}_i$: Maximum discharge rate in powerhouse i [m^3/s].

$\bar{V}_i$: Maximum water spill in powerhouse i [m^3/s].

$\bar{W}_t^+$: Maximum electricity sale at time t [MWh/day].

$\bar{W}_t^-$: Maximum electricity purchase at time t [MWh/day].

Derived sets

$\mathcal{T}(m)$: Time periods when maintenance task m can be initiated in order to be completed within $\mathcal{T}$.

$\mathcal{M}(i)$: Maintenance tasks m that should be executed in powerhouse i.

$\mathcal{M}(i,t)$: Maintenance tasks m that can be in execution in powerhouse i at time period t.

$\mathcal{U}(i)$: Powerhouses upstream of powerhouse i ($\mathcal{U}(i) \subset \mathcal{I}$).

$\mathcal{K}(i,t)$: Numbers of generators that can be active at time period t and powerhouse i.

$\mathcal{H}(i,k)$: Hyperplanes for approximating the maximum power output of powerhouse i when k turbines are active.

Parameters with indexes in derived sets

β_h^u : Coefficient of u_{it} in hyperplane $h \in \mathcal{H}(i,k)$ for bounding the power output of powerhouse i when k generators are active [MWh· s/(m^3·day)].

β_h^s : Coefficient of s_{it} in hyperplane $h \in \mathcal{H}(i,k)$ for bounding the power output of powerhouse i when k generators are active [MWh/(hm^3·day)].

β_h^0 : Independent term of hyperplane $h \in \mathcal{H}(i,k)$ for bounding the power output of powerhouse i when k generators are active [MWh/day].

Decision variables

r_{it} : Number of maintenance activities in execution at powerhouse i and time period t.

p_{it} : Corrected estimate of the electricity production of powerhouse i during time period t [MWh/day].

$\hat{p}_{it}$: Outer approximation of the electricity production in of powerhouse i during time period t [MWh/day].

p_{itk} : Estimate of the electricity production in powerhouse i during time period t when k generators are active [MWh/day].

s_{it} : Content of reservoir in powerhouse i at the end of period t [hm^3].

v_{it} : Water spill of reservoir in powerhouse i at time period t [m^3/s].

u_{it} : Water discharge of turbines in powerhouse i at time period t [m^3/s].

w_t^-, w_t^+ : Purchase and sale of electricity, respectively, at period t [MWh].

y_{mt} : Binary variable with value 1 if maintenance task m initiates at time period t, 0 otherwise.

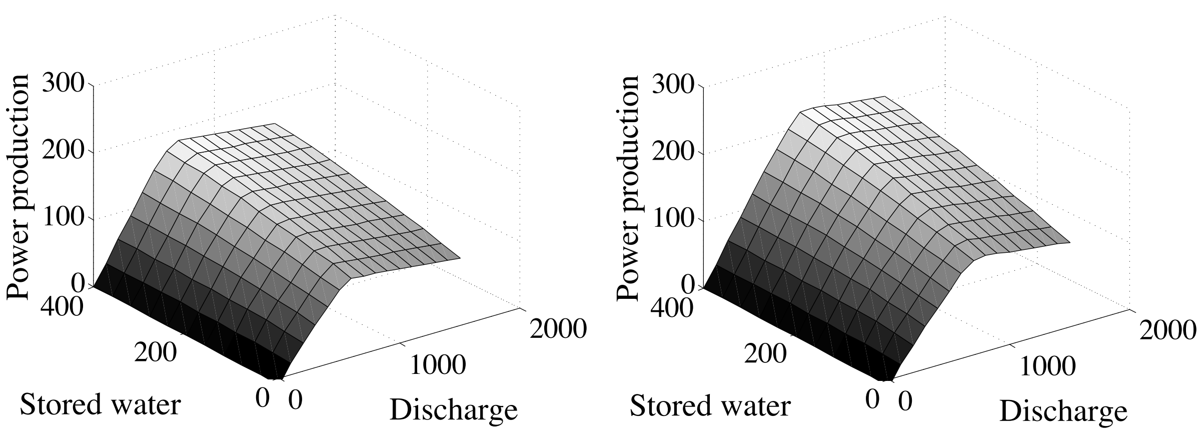
z_{itk} : Binary variable with value 1 if k hydro-turbines are active in powerhouse i at time t, 0 otherwise.

4.2 Introduction

In the power industry, preventive maintenance activities are carried out on a regular basis to prevent expensive equipment failures and to ensure a continuous operation within acceptable efficiency levels. As generators under maintenance are typically inactive, turbine discharges, water spill and electricity production are affected by maintenance activities. Therefore, the maintenance scheduler should anticipate the impact of the maintenance plan on the system operation cost. However, in hydroelectric systems these costs are difficult to estimate due to multiple interrelated physical variables. In particular, hydroelectricity production is a function of both the potential energy (the water head) and the kinetic energy of the water that drives the turbine-generators of the system. The relationship between these variables is defined by the Hydropower Production Function (HPF)

$$p = \rho g \gamma q h \eta(q, h), \tag{4.1}$$

where p is the power output (MW), ρ the water density (kg/m^3), g the gravitational acceleration (m/s^2), γ the conversion factor (10^{-6}), q the turbine water discharge (m^3/s), h the net water head (m), and $\eta(q, h)$ the turbine-generator efficiency (%). For each turbine the efficiency η is a nonlinear function of the net water head and the water discharge of the turbine. Therefore, the efficiency factor η introduces further nonlinearities in the power production of the system.
As the set of active generators affects the generation capacity as well as the optimal quantities of water spill and water discharge, the number of active generators has a nonlinear effect on the total power output. Fig. 4.1 shows the power production as a function of water discharge and stored water in a reservoir for either four or five active generators.



(a) Four generators

(b) Five generators

Figure 4.1 The maximum power output as a function of water discharge and stored water varies according to the number of active generators

Spatial and temporal inter-dependencies should also be considered in the hydropower operation. First, because water discharges can feed downstream turbines in the current or in subsequent time periods, and second, because future operation costs are determined by present decisions, such as generator outages and water spills from reservoirs. All the aforementioned elements make the optimal planning of maintenance outages in hydropower systems a challenging endeavor.

In the electricity industry, the Generator Maintenance Scheduling Problem (GMSP) has been widely studied, see e.g. Froger et al. (2016). However, specific operating conditions of hydroelectric systems have been scarcely addressed in GMSP. In a GMSP, Feng et al. (2011) represented the uncertainty of the power output with fuzzy variables, but omitted water storage levels and water head effects. Foong et al. (2008) proposed a meta-heuristic for a GMSP with an oversimplified hydropower operation that considers constant power output in active units. Kuzle et al. (2010) introduced transmission constraints in a simple GMSP where the nonlinearity of the production functions is neglected. Likewise, Perez-Canto (2008) omitted relevant characteristics of hydropower systems, such as temporal and spatial interdependencies, and nonlinearities of the power production. Clearly, a finer representation of the hydropower system's characteristics is necessary to achieve valid solutions to the GMSP for hydropower systems in practice.

Several works have considered the nonlinearity of the HPF (4.1) for short-term hydropower operation, without incorporating maintenance scheduling decisions (Finardi and da Silva, 2006; Arce, 2001; Catalão et al., 2009; Conejo et al., 2002; Borghetti et al., 2008; Diniz and Maceira, 2008; Séguin et al., 2016). Finardi and da Silva (2006), Arce (2001) and Catalão et al. (2009) used nonlinear functions for estimating the power production of hydro units. However, as the nonlinearity of the HPF makes hydrower scheduling problems hard to solve, different linear approximation approaches have been proposed. For the day-ahead scheduling of generators, Conejo et al. (2002) introduced piecewise linearization for representing the effects of the water discharge on the power production. The water head effect on the power output was estimated by interpolation among piecewise approximations for different stored water levels. Following a similar approach, Borghetti et al. (2008) proposed a refined linearization for representing the water head effects in hydro unit commitment. Due to the size of the resulting model, results were only reported for a single-reservoir system. For the short-term hydrothermal dispatch problem, Diniz and Maceira (2008) approximated with linear inequalities the HPF (4.1), considering the effects of water spill and water head. More recently, Séguin et al. (2016) approximated the power output with smoothing splines for the short-term scheduling of hydro units. These splines were fitted to a maximum power output surface computed by means of dynamic programming for different values of water discharge and stored water level in a reservoir.

In this paper, we propose a mixed-integer linear programming model for the GMSP in hydropower systems that accounts for the nonlinearities of hydroelectric operations via a convex hull approximation of the hydropower production function. Given the difficulty of the resulting optimization problem, we explore three approaches for strengthening the formulation: extended formulation, set reduction, and valid inequalities. The set reduction uses logical conditions for excluding superfluous set elements, in order to reduce the variables and constraints of the model. The possible combinations of these approaches lead to eight formulations that we compared in terms of computational times and optimality gaps on test instances adapted from a real hydropower system in Canada.

This paper is structured as follows. Section II presents our basic mixed integer programming mathematical model. Section III describes the approaches to improve the formulation and the resulting alternative formulations. Section IV reports our computational experiments for evaluating the different alternatives. Section V summarizes our findings and concludes the paper.

4.3 A basic mixed integer programming formulation

We consider the GMSP for hydropower systems in the general form

$$\max_{\substack{y \in \mathcal{Y} \\ x(y) \in \mathcal{X}(y)}} \quad \Phi(x(y)) - \Psi(y), \tag{4.2}$$

where the variables y denote the maintenance decisions and the variables $x(y)$ represent operational decisions, such as turbine discharges and water spills. The feasible set $\mathcal{X}(y)$ of the operational decisions is determined by the water balance constraints and the bounds of the hydropower operation, which are affected by the scheduled outages y. The set $\mathcal{Y}$ of feasible maintenance decisions is defined by the time window constraints of the maintenance activities, the maximum number of simultaneous maintenance outages, and other logical constraints. In (4.2), the functions $\Psi(y)$ and $\Phi(x(y))$ denote the maintenance cost and the value of the electricity production during the planning horizon, respectively. Note that the value of the electricity production $\Phi(x(y))$ is affected by the maintenance schedule y because the power production function is different for each set of active generators (Fig. 4.1). The interdependency between the maintenance plan and the hydropower operation makes this a challenging nonlinear, nonconvex and combinatorial optimization problem.

In the next subsections we formulate in turn the hydropower operation, the linear approximation of the power production function, and the maintenance scheduling.

4.3.1 The hydropower operation

The hydropower operation problem optimizes the water discharges, water spills and stored water levels to maximize the total expected value of the electricity production, while respecting the physical constraints of the system and the target levels of the reservoirs at the end of the planning horizon. The physical constraints enforce the mass and power balance, as well as the bounds of the variables, such as the water levels in reservoirs. At each time period $t \in \mathcal{T}$, reservoirs can be fed by lateral inflows F_{it} from tributary rivers or snow-melt, or by turbine discharges u_{gt} and water spills v_{gt} from upstream reservoirs $g \in \mathcal{U}(i)$.

At each powerhouse and time period, the mass balance (4.3) implies that the initial water volume $s_{i(t-1)}$ minus the water volume s_{it} at the end of the time period should be equal to the water inflows minus the total outflows, multiplied by the conversion factor Q. As it is customary, we assume that the outflows are equal to the total turbine discharge u_{it} and the water spill v_{it} of the reservoir.

$$s_{it} - s_{i(t-1)} = Q\left(F_{it} + \sum_{g \in \mathcal{U}(i)} [u_{gt} + v_{gt}] - u_{it} - v_{it}\right), \quad \forall\, t \in \mathcal{T},\, i \in \mathcal{I}. \tag{4.3}$$

To ensure the consistency with the initial and the final water volume of the reservoirs, we define $s_{i(t-1)} = S_{i0}$ for $t = 1$ in (4.3). In addition, (4.4)-(4.6) define the bounds on the water discharge, water spill and water volume.

$$0 \leq u_{it} \leq \bar{U}_i, \quad \forall i \in \mathcal{I},\, t \in \mathcal{T}, \tag{4.4}$$

$$0 \leq v_{it} \leq \bar{V}_i, \quad \forall i \in \mathcal{I},\, t \in \mathcal{T}, \tag{4.5}$$

$$\underline{S}_i \leq s_{it} \leq \bar{S}_i, \quad \forall\; i \in \mathcal{I},\, t \in \mathcal{T}. \tag{4.6}$$

The energy balance (4.7) requires that at each time period t, the total energy production plus the energy purchases equal the load J_t plus the energy sales:

$$\sum_{i \in \mathcal{I}} p_{it} + w_t^- = J_t + w_t^+, \quad \forall\; t \in \mathcal{T}, \tag{4.7}$$

with bounded electricity trade variables,

$$0 \leq w_t^+ \leq \bar{W}^+, \quad \forall\; t \in \mathcal{T}, \tag{4.8}$$

$$0 \leq w_t^- \leq \bar{W}^-, \quad \forall\; t \in \mathcal{T}. \tag{4.9}$$

Notice that this definition of the power balance in (4.7) can describe a variety of situations for electricity producers. For example, the parameter $J_t > 0$ can represent the case of a producer that in a liberalized electricity market has committed to supply an amount of electricity J_t in the forward market, and that in the spot or day ahead market can trade electricity (w_t^-, w_t^+)

to compensate for the differences between its forward commitment J_t and its actual electricity production. Clearly, if at some time period $t \in \mathcal{T}$ the electricity purchase is not allowed, it suffices to define $w_t^- = 0$.

4.3.2 Linearization of the power production function

As the nonlinearity of the electricity production functions (Fig. 4.1) poses a challenge to the solution of the GMSP, we approximate these functions with linear inequalities. In this way, we can formulate the GMSP as a mixed-integer linear program (MILP), which can be tackled with state-of-the art solvers (Bixby and Rothberg, 2007).
For each powerhouse, the power output p_{it} is a nonlinear function Θ_i of the water discharge u_{it} and the net water head (which in turn is a nonlinear function of the stored water volume s_{it} and the total water discharge u_{it}). Since each generator may have a particular efficiency curve, the maximum power output in a powerhouse depends on the specific set of active generators. However, if the differences among power functions of individual generators are negligible, the power function in a powerhouse can be characterized by the number of active generators k_{it}, instead of the explicit set of active generators, that is, $p_{it} = \Theta_i(s_{it}, u_{it}, k_{it})$. This assumption significantly reduces the problem complexity, since otherwise a specific power function would be necessary for each combination of active generators.

For each number of active generators with their respective efficiency curves, a dynamic programming algorithm can determine the commitment of units, as well as the maximum power output corresponding to a set of water discharges and stored water levels (Fig. 4.1) (Séguin et al., 2016). By definition, this set of points is contained in its convex hull, whose half-space representation can be obtained with a facet enumeration algorithm. Some implementations of this algorithm are freely available (Papazafeiropoulos, 2014; Fukuda, 2011).

The resulting polyhedron may contain a large number of hyperplanes, some of which should be dropped since they define the lower facets of the convex hull with respect to the power output p_{it}. The set can be additionally reduced by iteratively removing the hyperplane for which the remaining polyhedron has the smallest approximation error of the power output. This sequential elimination of hyperplanes is repeated until the target number of hyperplanes or a specified precision is reached.

For each powerhouse i and number of active generators k, the resulting set of hyperplanes $\mathcal{H}(i,k)$ with parameters β_h^0, β_h^u and β_h^s provides an outer approximation $\hat{p}_{it}$ of the power output corresponding to the specific amounts of water discharge u_{it} and stored water level

s_{it}, when k generators are active, i.e.,

$$0 \le \hat{p}_{it} \le \beta_h^0 + \beta_h^u u_{it} + \beta_h^s s_{it}\,,$$
$$\forall\ i \in \mathcal{I},\ t \in \mathcal{T},\ k \in \mathcal{K}(i,t),\ h \in \mathcal{H}(i,k).$$

Notice that through the index $h \in \mathcal{H}(i,k)$, the hyperplane parameters β_h^0, β_h^u and β_h^s are defined for the corresponding powerhouse i and number of active generators k.
At powerhouse i and time period t, if k^* is the number of active generators, the power function constraints for $k \neq k^*$ can be relaxed by adding the bounding term $(1 - z_{itk})\bar{P}_i$ on the right hand side of (4.10), i.e.,

$$\begin{aligned} 0 \le \hat{p}_{it} \le \beta_h^0 + \beta_h^u u_{it} + \beta_h^s s_{it} + (1 - z_{itk})\bar{P}_i, \\ \forall\ i \in \mathcal{I},\ t \in \mathcal{T},\ k \in \mathcal{K}(i,t),\ h \in \mathcal{H}(i,k), \end{aligned} \tag{4.10}$$

where $\bar{P}_i$ is the generation capacity of powerhouse i and the binary variables z_{itk} indicate if k generators are active at (i,t). Since only one binary variable z_{itk} takes value 1 for each $(i,t) \in \mathcal{I} \times \mathcal{T}$,

$$\sum_{k \in \mathcal{K}(i,t)} z_{itk} = 1,\ \forall\ i \in \mathcal{I}, t \in \mathcal{T}. \tag{4.11}$$

Thus, by (4.11) and the binary condition on z_{itk}, the approximated power output $\hat{p}_{it}$ in (4.10) is bounded only by the hyper-plane set corresponding to the number of active turbines.
The quality of the approximation given by (4.10) increases with the number of hyperplanes in $\mathcal{H}(i,k)$ and with the convexity of the actual power production function. Thus, there is a compromise between model size and solution quality. In our tests with real data the approximation errors of this approach were 0.5% and 0.25% of the electricity production for 15 and 30 hyperplanes in $\mathcal{H}(i,k)$, respectively. Nevertheless, the overestimate of the power production can be reduced with

$$p_{it} = \alpha_0 + \alpha_1 \hat{p}_{it},\ \forall\ i \in \mathcal{I}, t \in \mathcal{T}, \tag{4.12}$$

where p_{it} is the corrected estimate of the electricity production and α_0, α_1 are the parameters of a linear regression model that fits the estimated electricity production $\hat{p}_{it}$ to the corresponding amounts of actual electricity production, using historical data.

4.3.3 The maintenance scheduling problem

For each maintenance activity $m \in \mathcal{M}$, the interval between the earliest starting time period E_m and the latest starting time period L_m defines the set of time periods $\mathcal{T}(m)$ when the activity m can start: $\mathcal{T}(m) = \{t \in \mathcal{T} \mid E_m \leq t \leq L_m\}$. We assume that each activity can be completed within the planning horizon $\mathcal{T}$, i.e., $E_m \leq L_m \leq T - D_m + 1$, where D_m denotes the duration of the maintenance task m.

The definition of the binary variables y_{mt}, $\forall\, m \in \mathcal{M}, t \in \mathcal{T}(m)$, for representing the maintenance decisions (see notation in Section 4.1) avoids the definition of time window constraints since the set $\mathcal{T}(m)$ encodes the time window parameters of each activity. Unnecessary y_{mt} variables are excluded from the model because they are defined using $\mathcal{T}(m)$ instead of $\mathcal{T}$.

For the basic maintenance problem we consider only the constraints on: completion of maintenance tasks, maximum number of generator outages, and mapping the maintenance schedule to the number of active generators.

The task completion constraints (4.13) enforce each activity to start at one of the feasible time periods $\mathcal{T}(m)$. Constraints (4.14) compute for each powerhouse the number of maintenance activities r_{it} in execution at time period t, among the set of activities $\mathcal{M}(i)$ that must be completed at station i.

$$\sum_{t \in \mathcal{T}(m)} y_{mt} = 1, \quad \forall\, m \in \mathcal{M}. \tag{4.13}$$

$$\sum_{\substack{m \in \mathcal{M}(i) \\ t' \in \{\mathcal{T}(m) \mid (t - D_m + 1) \leq t' \leq t\}}} y_{mt'} = r_{it}, \quad \forall\, i \in \mathcal{I}, t \in \mathcal{T}. \tag{4.14}$$

Notice that at time period t an activity m is in execution if it starts between $t - D_m + 1$ and t. This is the interval of index t' on the summation term in (4.14).

The maximum number of outages O_{it} bounds r_{it}:

$$0 \leq r_{it} \leq O_{it}, \quad \forall\, i \in \mathcal{I}, t \in \mathcal{T}. \tag{4.15}$$

O_{it} depends on the maintenance resources. In addition, for a feasible operation, O_{it} cannot exceed the difference between the number of available generators $\bar{G}_{it}$ and the minimum number of generators in service $\underline{G}_i$, i.e., $O_{it} \leq \bar{G}_{it} - \underline{G}_i$, $\forall\, i \in \mathcal{I}, t \in \mathcal{T}$. Notice that $\bar{G}_{it}$ is a time varying parameter, since the number of available generators can be affected by existing generator outages or by previous maintenance scheduling decisions. On the other hand, the minimum number of generators $\underline{G}_i$ is constant in time due to operational requirements.

Constraints (4.16) map the number of outages r_{it} into the variables z_{itk}. At each period and powerhouse, the maximum number of available generators $\bar{G}_{it}$ equals the sum of the number of outages r_{it} plus the number of active generators k^* corresponding to $z_{itk^*} = 1$.

$$r_{it} + \sum_{k \in \mathcal{K}(i,t)} k z_{itk} = \bar{G}_{it}, \forall\, i \in \mathcal{I}, t \in \mathcal{T}. \tag{4.16}$$

Constraints (4.18)-(4.17) specify the binary decision variables.

$$z_{itk} \in \{0,1\},\ \forall\, i \in \mathcal{I}, t \in \mathcal{T},\ k \in \mathcal{K}(i,t), \tag{4.17}$$

$$y_{mt} \in \{0,1\},\ \forall m \in \mathcal{M},\ i \in \mathcal{T}(m). \tag{4.18}$$

4.3.4 The objective function

The GMSP maximizes the value of the electricity production plus the value of the stored water, minus the sum of the maintenance costs:

$$\underset{\substack{w^+,w^-,u,v,s,\\ p,y,z}}{\text{Maximize}} \sum_{t \in \mathcal{T}} \left(B_t^+ w_t^+ - B_t^- w_t^- \right) - \sum_{\substack{m \in \mathcal{M},\\ t \in \mathcal{T}(m)}} C_{mt} y_{mt}. \tag{4.19}$$

The value of the electricity production during the planning horizon is calculated as the net benefit of the electricity trade, i.e., the difference between the revenue of electricity sale ($B_t^+ w_t^+$) and the cost of electricity purchase ($B_t^- w_t^-$).

4.3.5 The complete basic model

We refer to the resulting mixed-integer linear programming (MILP) problem as P_B:

Maximize (4.19) subject to constraints (4.3) - (4.18).

Notice that for any feasible maintenance schedule $(\bar{y}, \bar{z})$, the resulting hydropower operation subproblem P_H is the linear program

$$P_H(\bar{y}, \bar{z}) = \underset{w^+,w^-,u,v,s,p}{\text{Maximize}} \sum_{t \in \mathcal{T}} \left(B_t^+ w_t^+ - B_t^- w_t^- \right), \tag{4.20}$$

subject to (4.3) -(4.10), (4.12).

Naturally, in P_H the simultaneous purchase and sale of electricity (i.e., the case of arbitrage) can be prevented if the sale price of the electricity B_t^+ is lower than the purchase price B_t^- as stated in Proposition 1.

Proposition 1. *In any optimal solution to $P_H(\bar{y}, \bar{z})$ with electricity prices $B_t^+ < B_t^-$, either $w_t^+ = 0$ or $w_t^- = 0$.*

See Appendix 4.8.1 for a proof of this proposition.

Furthermore, this property also holds for any feasible solution to P_B obtained with a general MILP solver (e.g., CPLEX Gurobi, Xpress-MP), even if the maintenance schedule is not optimal. Indeed, any feasible solution returned by such a solver is obtained at a node of the search tree by solving a linear program to optimality.

4.4 Tightening approaches

Due to the weak continuous relaxation of (4.10) and (4.16), the formulation in Section 4.3.5 is difficult to solve for realistic instances. In this section we explore three approaches for tightening the formulation: extended formulation, set reduction and valid inequalities.

4.4.1 Extended formulation

The bound (4.10) can be very weak because it is valid for any operating condition and for any number of active generators k on the interval $[\underline{G}_i, \bar{G}_{it}]$. However, $\bar{P}_{ik}$ and p_{itk} can be based on the actual number of active generators k and the specific operating conditions at each time period and powerhouse. Constraints (4.21) specify the power bound for each number of active generators, and (4.22) ensure the equivalence with the original variables p_{it} and in substitution of (4.10), constraints (4.23) define a linear approximation of the power function.

$$0 \leq p_{itk} \leq z_{itk} \bar{P}_{ik}, \quad \forall\, i \in \mathcal{I},\, t \in \mathcal{T},\, k \in \mathcal{K}(i,t). \tag{4.21}$$

$$\sum_{k \in \mathcal{K}(i,t)} p_{itk} = p_{it}, \quad \forall\; i \in \mathcal{I},\, t \in \mathcal{T}, \tag{4.22}$$

$$\begin{aligned} 0 \leq p_{itk} \leq \beta_h^0 + \beta_h^u u_{it} + \beta_h^s s_{it}, \\ \forall\, i \in \mathcal{I},\, t \in \mathcal{T},\, k \in \mathcal{K}(i,t), h \in \mathcal{H}(i,k). \end{aligned} \tag{4.23}$$

Thus we have P_E as the MILP with the extended formulation:

Maximize (4.19) subject to (4.3)-(5.15), (4.11)-(4.18), (4.21)-(4.23).

The bounds $\bar{P}_{ik}$ for (4.21) can be obtained as the optimal values q^*_{ik} from maximizing the power output in (4.23) when the stored water level is maximum:

$$\underset{q,u}{\text{Maximize}}\ q_{ik} \text{ s.t. } q_{ik} \leq \beta^0_h + \beta^u_h u_{itk} + \beta^s_h \bar{S}_i,\ \forall\, h \in \mathcal{H}(i,k). \tag{4.24}$$

4.4.2 Set reduction

Next we exploit the time window parameters of the maintenance tasks in order to exclude unnecessary set elements. As a consequence, fewer constraints and variables are defined, leading to a tighter continuous relaxation and fewer choices for branching. We aim at reducing the set $\mathcal{K}(i,t)$ that determines both the number of binary variables z_{itk} and the degrees of freedom of the system (4.11) and (4.16). A maintenance activity m beginning at E_m and with duration D_m spans the interval $\mathcal{T}^E(m) = \{\, t \in \mathcal{T}(m) \,|\, E_m \leq t < E_m + D_m\}$. Likewise, if activity m starts at L_m, it spans the interval $\mathcal{T}^L(m) = \{\, t \in \mathcal{T}(m) \,|\, L_m \leq t < L_m + D_m\}$. The overlap of the two intervals

$$\begin{aligned}\mathcal{T}^{\mathcal{O}}(m) &\triangleq \mathcal{T}^E(m) \cap \mathcal{T}^L(m)\\ &= \{\, t \in \mathcal{T}(m) \,|\, L_m \leq t < E_m + D_m\},\end{aligned}$$

defines the set of time periods when the activity necessarily will take place. Likewise, the span of a maintenance activity m is the interval $\mathcal{T}^{\mathcal{S}}(m)$ where the activity can be in execution. Since activity m cannot start before E_m and it must finish before $L_m + D_m$, we define

$$\mathcal{T}^{\mathcal{S}}(m) = \{\, t \in \mathcal{T}(m) \,|\, E_m \leq t < L_m + D_m\}.$$

These definitions are illustrated in Fig. 4.2.

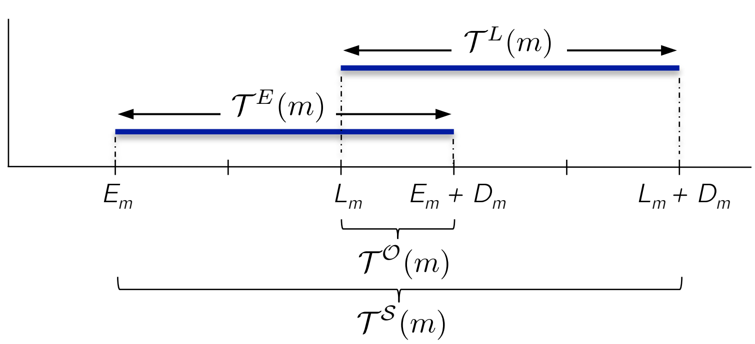


Figure 4.2 Timeline for a maintenance activity m.

The maximum number of maintenance activities in execution at powerhouse i during time period t is the cardinality of the set of tasks whose spans $\mathcal{T}^{\mathcal{S}}(m)$ intersect at time period t, that is,

$$\bar{R}_{it} = |\{\, m \in \mathcal{M}(i) \,|\, t \in \mathcal{T}^{\mathcal{S}}(m) \,\}|.$$

Similarly, the minimum number of activities in execution at powerhouse i during time period t is,

$$\underline{R}_{it} = |\{\, m \in \mathcal{M}(i) \,|\, t \in \mathcal{T}^{\mathcal{O}}(m) \,\}|.$$

Naturally, $\bar{R}_{it}$ and $\underline{R}_{it}$ bound the number of outages r_{it}:

$$\underline{R}_{it} \leq r_{it} \leq \bar{R}_{it}, \;\; \forall\, i \in \mathcal{I}, t \in \mathcal{T}. \tag{4.25}$$

Proposition 2. *In formulations P_B and P_E, the feasible number of active generators k at period $t \in \mathcal{T}$ and powerhouse $i \in \mathcal{I}$ is in the set*

$$\mathcal{K}(i,t) = \Big\{\, k \in \mathbb{Z} \,|\, \underline{K}_{it} \leq k \leq \bar{K}_{it} \Big\}, \tag{4.26}$$

where

$$\underline{K}_{it} = \max\{\bar{G}_{it} - O_{it},\, \bar{G}_{it} - \bar{R}_{it}\}, \tag{4.27}$$

$$\bar{K}_{it} = \bar{G}_{it} - \underline{R}_{it}. \tag{4.28}$$

See Appendix 4.8.2 for a proof of this proposition.

From (4.26-4.28) we see that the greater the difference between $\bar{G}_{it}$ and $\bar{K}_{it}$, as well as between $\underline{G}_i$ and $\underline{K}_{it}$, the greater the reduction in the number of variables and constraints with index $k \in \mathcal{K}(i,t)$.

4.4.3 Valid inequalities

Finally, we analyze the linear system formed by constraints (4.11) and (4.16), which in general is undetermined and has multiple non-integer solutions. We consider the case when $\underline{R}_{it} = 0$.

If $r_{it} = 0$, then from constraints (4.16), $\sum_{k \in \mathcal{K}(i,t)} z_{itk} k = \bar{G}_{it}$, which implies $z_{itk} = 1$ for $k = \bar{G}_{it}$, since by constraint (4.11) only one binary variable z_{itk} should be active for each $(i,t) \in \mathcal{I} \times \mathcal{T}$. On the other hand, if $r_{it} \geq 1$, then $z_{itk} = 0$ for $k = \bar{G}_{it}$ with $(i,t) \in \mathcal{I} \times \mathcal{T}$. By disaggregating r_{it} into the corresponding y_{mt} variables (see (4.14)), these logical implications are equivalent to

$$\sum_{t' \in \{\mathcal{T}(m) \,|\, (t-D_m+1) \le t' \le t\}} y_{mt'} + z_{itk} \le 1, \quad \text{for } k = \bar{G}_{it}, \qquad \forall\, i \in \mathcal{I},\, m \in \mathcal{M}(i),\, t \in \mathcal{T}, \tag{4.29}$$

which by the binary condition on z_{itk} and y_{mt} are facet defining inequalities.

Also, since $r_{it} = 0$ implies $z_{itk} = 0 \;\forall\, k \in \{\mathcal{K}(i,t) \setminus \bar{G}_{it}\}$,

$$\sum_{k \in \mathcal{K}(i,t) \setminus \bar{G}_{it}} z_{itk} \le r_{it}, \;\forall\, i \in \mathcal{I},\, t \in \mathcal{T}. \tag{4.30}$$

Next we show that constraints (4.29)-(4.30) allow relaxing the integrality of a subset of binary variables z_{itk} when $\bar{K}_{it} = \bar{G}_{it}$ and the number of degrees of freedom of the system (4.11),(4.16) is sufficiently small.

Proposition 3. *In models P_B and P_E with constraints* (4.29)-(4.30) *if for some* $(i', t') \in \mathcal{I} \times \mathcal{T}$*:*

i) $\underline{R}_{i't'} = 0$*,*

ii) $\bar{K}_{i't'} - \underline{K}_{i't'} \le 2$*,*

iii) there exists an integer feasible solution,

then the integrality condition (4.17) *for* $z_{i't'k} \;\forall\, k \in \mathcal{K}(i', t')$ *can be relaxed as the variables* $z_{i't'k}$ *will be integer in any feasible solution.*

See Appendix 4.8.3 for a proof of this proposition.

4.5 Computational experiments

In this section we report on our computational experiments to evaluate the eight formulations obtained starting from the basic model and including/excluding each of the three approaches in Section 4.4. The eight combinations are given in Table 4.2, where 1 indicates that a given approach is used in the corresponding formulation, and 0 indicates that the approach was not used.

We conducted two experiments to determine the best combination. First, we solved smaller instances of the GMSP and we analyzed the computation times in order to select a subset of formulations. Second, we evaluated this subset via experiments with larger instances.

Our test instances were adapted from a cascade 4-powerhouse system. For each powerhouse and number of generators, we approximated the hydropower production function with 30 linear inequality constraints (4.10) and (4.23). For each instance, maintenance requirements are specified with the following parameters for each activity: index, powerhouse, duration, earliest start time period, and latest start time period. We maximize the value of the electricity production, with a sale price of 8 \$/kWh, and $J_t = 0$ and $w_t^- = 0, \forall t \in \mathcal{T}$.

4.5.1 Computational results for all formulations

For the first experiment, we defined two levels for each of the five factors of the instance size (Table 4.1). For each of the $2^5 = 32$ combinations of these factors, we created two

Table 4.1 Levels of factors used to create the test intances to compare all formulations in Section 4.5.1.

Factor	Low Level	High Level
Number of maintenance tasks	8	10
Number of time periods	20	25
Time window length	5	8
Maximum outages in each powerhouse	2	3
Avg. duration of maintenance tasks	4	5

maintenance datasets, for a total of 64 test instances. The size of the MILP formulations ranged from 94 binary variables, 390 continuous variables and 4263 constraints, to 456 binary variables, 775 continuous variables and 12485 constraints. Because randomly generating instances for the GMSP is prone to infeasibilities, we created new instances with random changes in a subset of parameters of initial feasible instances. When an infeasible instance was obtained by this procedure, we restored its feasibility by arbitrarily changing the instance parameters.

We ran the tests in a 24-processor Intel® Xeon® server at 2.7 GHz with 32.9 GB RAM, with 4 cores dedicated for running the Xpress-MP solver. The models were coded in C^{++} with the Xpress BCL 8.1.0 callable library (FICO, 2014).

We chose CPU clock time as the basic performance metric, which allows to measure the actual computation time for solving the problem, without the effect of background processes. Given that the computation times increase significantly with the size of the instance and also differ between instances of similar size, we normalized for each instance the logarithmic CPU time according to the standard score

$$z_{jb} = (t_{jb} - \mu_j^t)/\sigma_j^t, \tag{4.31}$$

where t_{jb} is the logarithmic CPU time for solving instance $j \in \mathcal{J}$ with formulation $b \in \mathcal{B}$, and μ_j^t, σ_j^t are respectively the mean and the standard deviation of the logarithmic CPU times of the 8 models for solving instance j.

We report in Table 4.2 the mean $\bar{z}_b$ and standard deviation σ_b^z of z_{jb} over the 64 test instances, for each formulation. The results show that the choice of formulation affects the computation times, as corroborated with a p-value of 0.005 for a one-way ANOVA, which for a significance level of $\alpha = 0.01$ indicates a significant effect of the selected formulation on the logarithmic CPU time.

Table 4.2 Normalized log CPU times per instance, computed from 64 test instances.

Formu-	Tightening approaches			Norm. log CPU time	
lation	Set reduc.	Valid ineq.	Extended formul.	Average $\bar{z}_b$	St. dev. σ_b^z
1	0	0	0	**1.469**	0.35
2	0	0	1	-0.849	0.40
3	0	1	0	0.790	0.38
4	0	1	1	-0.685	0.33
5	1	0	0	0.421	**0.50**
6	**1**	**0**	**1**	**-0.880**	**0.34**
7	1	1	0	0.511	0.39
8	1	1	1	-0.776	0.42

In these instances, the wall-clock time to reach optimality ranged from 1 s to 1743 s, with an average of 84.27 s over all formulations. The computational wall-clock time was highly correlated with the CPU time ($R^2 = 0.99$).

While formulation 1 had the largest average normalized log CPU time, the smallest time was achieved by formulation 6 (extended formulation with set reduction). The latter also had the second smallest standard deviation. The maximum standard deviation corresponded to the formulation with only set reduction. Overall, the formulations 2, 4, 6, and 8 gave the best results in Table 4.2. In several instances, we registered more than one order of magnitude of difference in wall-clock time between the basic model (formulation 1) and the best formulation. However, in Table 4.2 these differences are attenuated by the logarithmic transformation that we applied.

The effect of the choice of formulation also shows in the performance profiles of Fig. 4.3. A performance profile (Dolan and Moré, 2002) gives the cumulative relative frequency $\rho_b(\tau)$ with which a formulation solves instances of the problem within a factor τ of the best possible value of $\log_2(r_{jb})$, where $r_{jb} = t_{jb}/\min_{b \in B} t_{jb}$, and

$$\rho_b(\tau) = \frac{1}{n_j} \text{size}\{j \in \mathcal{J} : \log_2(r_{jb}) \leq \tau\}. \tag{4.32}$$

In summary, the curves closest to the top left corner correspond to the formulation with the best performance.

Fig. 4.3 shows that the formulations with at least one tightening component perform better than the basic model (formulation 1). In Fig. 4.3, the performance profiles of the best 4 formulations indicate that formulation 6 is a clear winner for $\tau \leq 0.8$. In less than 10% of the instances, models 2 and 8 are a competitive choice.

The extended formulation is common to the 4 best-performing formulations in Fig. 4.3. The ANOVA results in Table 4.3 show that this approach, either alone or in combination with others, has a significant effect for arbitrarily small significance α levels (p-value = 2.36e-12). On the other hand, although the formulation with only valid inequalities outperformed the basic model (formulation 3 vs. formulation 1 in Table 4.2 and Fig. 4.3), the effect of the valid inequalities was not statistically significant when combined with other formulation approaches (p-value = 0.758). Finally, the effect of set reduction is only significant for $\alpha \geq 0.2$ (p-value = 0.181).

Table 4.3 p-values based on normalized log CPU time

Approach	p-value
Set reduction	0.181
Valid inequalities	0.758
Extended formulation	2.36e-12

4.5.2 Optimality gaps of the best formulations

In the second experiment, we worked only with formulations 2, 6 and 8. These have the smallest average CPU times in Table 4.2, and clearly outperform formulation 4 in Fig. 4.3. Our focus is on the optimality gaps that these formulations can achieve for large instances of the GMSP. We tested these formulations on 16 instances with more maintenance tasks than the earlier instances. These 16 instances were generated with two maintenance datasets for

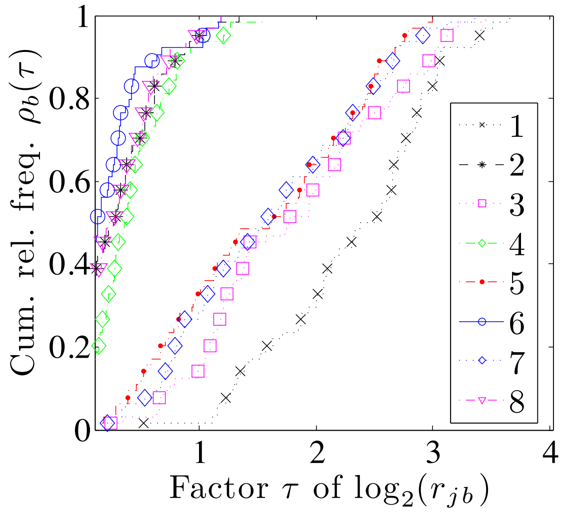


Figure 4.3 Performance profiles of the tested formulations

each of the $2^3 = 8$ combinations of the levels of the three factors in Table 4.4. For these

Table 4.4 Levels of factors for the test instances to compare the best formulations 4.5.2.

Factor	Low Level	High Level
Number of maintenance tasks	15	20
Time window length	5	8
Avg. duration of maintenance tasks	4	5

instances we specified a planning horizon with 25 time periods in a cascade 4-powerhouse system, with a maximum of 2 outages in each powerhouse.

Table 4.5 reports the optimality gap statistics for the three formulations after 1,000 and 20,000 seconds of CPU time on each instance.

All three formulations reached average optimality gaps below 3 % within 1,000 CPU s. Progress is substantially slower after that, and at the time limit of 20,000 CPU s the average optimality gap in all three formulations is close to 1.5%. Formulation 6 had the best overall performance after 1,000 CPU s, and formulations 2 and 6 had similar average performance after 20,000 CPU s. The average wall-clock time corresponding to the CPU time limit of 20,000 CPU s was 2,955.5 s, with a standard deviation of 51 s. Due to the specified time limit in this experiment, the optimal solution was not reached in any of the runs. However, the small optimality gaps in Table V indicate that with computational times beyond the specified time limit, the optimal solutions for the instance sizes that we considered are achievable in practice.

Table 4.5 Optimality gap statistics

Formulation	CPU time 20,000 s		CPU time 1,000 s	
	Mean	St. dev.	Mean	St. Dev
2	0.0144	0.0069	0.0295	0.0235
6	0.0144	0.0071	0.0229	0.0076
8	0.0151	0.0073	0.0273	0.0222

Based on the overall results, we conclude that the most promising approach is the extended formulation with set reduction (formulation 6), and possibly in combination with the valid inequalities.

4.6 Industrial application

We tested this approach with data adapted from a 4-powerhouse system of Rio Tinto in the Saguenay-Lac-St-Jean region in Québec, Canada (see Table 4.6). At the company, turbine-generator systems must undertake periodic preventive maintenance tasks of short duration. Less frequently, activities of longer duration, such as overhauling of generators, are also necessary. We considered 18 maintenance tasks to be completed in a planning horizon of 30 days. For each task, the time window, as well as the starting time of the activity according of an initial maintenance schedule are given. As in the previous section, the electricity production for each number of generators and powerhouse was approximated with 30 hyperplanes, and we set $J_t = 0$ and $w_t^- = 0, \forall\, t \in \mathcal{T}$. For this application, the relevant price is 5 ¢/kWh.

Table 4.6 Basic attributes of the hydropower system. Powerhouses are ordered from upstream to downstream.

System type	Number of generators	Installed capacity (MW)	Maintenance tasks
Reservoir	5	205	4
Run of the river	5	210	5
Reservoir	12	402	4
Run of the river	17	1587	5
Total	39	2404	18

We used formulation 6 (with set reduction and extended formulation) to solve this instance of the problem with the Xpress-MP solver in deterministic mode with 20 threads in a 24-

processor Intel® Xeon® server at 2.7 GHz with 32.9 GB RAM. As previous works on the GMSP (Feng et al., 2011; Foong et al., 2008; Kuzle et al., 2010; Perez-Canto, 2008) did not consider the maintenance time windows and other relevant aspects of the problem, these approaches can lead to infeasible solutions in practice. For this reason, in this industrial application example, we compare the value of the solution obtained with our model against the optimal maintenance schedule obtained with a simplified model P_S that neglects the nonlinearity of the electricity production, while still respecting the time windows of the maintenance tasks. Thus, we relax (4.23) to define P_S as

$$\text{Maximize (4.19) subject to (4.3)-(5.15), (4.11)-(4.18), (4.21)-(4.22).}$$

For the application example in this Section, the proposed model (formulation 6) has 7103 continuous variables, 299 binary variables and 7402 constraints. After 1207 s an optimal solution was found with an objective value of \$ 57.802 M. In contrast, the best solution found with the simplified model (P_S) has an objective value of \$ 67.444 M. The higher objective value of this solution is merely a consequence of the overestimated electricity production in P_S by ignoring the nonlinearity of the HPF. When the actual nonlinearity of the hydroelectricity production is considered, the maximum revenue of the maintenance schedule obtained with P_S is \$ 57.735 M. With respect to this solution, the optimal schedule of formulation 6 yields an increase of 1340 MWh of electricity production in the one-month planning horizon and an approximate annualized gain of \$ 804,000. The increment of the electricity production in the optimal solution is mainly a consequence of the reduction of accumulated water spills during the planning horizon, which translates into higher average stored water level and more efficient operation of the generators.

4.7 Conclusions

We proposed a mixed-integer optimization model for the GMSP in hydropower systems, and three possible approaches to tighten its continuous relaxation: set reduction, valid inequalities, and extended formulation. Using a set of 64 test instances, we found that the extended formulation had the most significant effect in decreasing the computational time, and that the combination of extended formulation and set reduction achieved the best average performance and small variability in computation time. This formulation was tested in a real 4-powerhouse hydropower system with 39 generators and 2404 MW of generation capacity, and an optimal maintenance schedule for a one-month planning horizon was found in less than 30 minutes.

We proved that under some conditions, the valid inequalities allow relaxing the integrality condition on a subset of binary variables of the problem. Although this insight did not exhibit a statistically significant effect in our tests, we consider that the mathematical result can be useful for developing heuristic solution methods for this problem as well as for other problems with similar integer-mapping constraints.

Because the GMSP typically spans a planning horizon of several weeks, in practice it may be possible to run the solver for several hours or even days, in order to obtain either optimal or near optimal solutions. However, more efficient solution methods are necessary to solve larger real instances. Furthermore, incorporating other relevant aspects of the problem, such as transmission system effects and uncertainty of water inflows will increase the computational complexity of the problem. Solution approaches considering these elements will be the subject of future work.

4.8 Appendices

4.8.1 Appendix A: Proof of proposition 1

Proof. By contradiction, suppose that $w_t^+ > 0$ and $w_t^- > 0$ for some t. Consider 3 cases: i) $w_t^+ > w_t^-$, ii) $w_t^- > w_t^+$, and iii) $w_t^- = w_t^+$. In case i), $B_t^- > B_t^+$ implies $-B_t^- w_t^- < -B_t^+ w_t^-$. Adding $B_t^+ w_t^+$ gives $B_t^+ w_t^+ - B_t^- w_t^- < B_t^+ w_t^+ - B_t^+ w_t^- = B_t^+(w_t^+ - w_t^-) = B_t^+ q_t^+$, which shows that $w_t^+ > w_t^- > 0$ is not optimal, since selling $q_t^+ = w_t^+ - w_t^-$ reaches higher profit than buying w_t^- and selling w_t^+. In case ii), $B_t^- > B_t^+$ implies $B_t^+ w_t^+ < B_t^- w_t^+$. Subtracting $B_t^- w_t^-$ gives $B_t^+ w_t^+ - B_t^- w_t^- < B_t^- w_t^+ - B_t^- w_t^- = B_t^-(w_t^+ - w_t^-) = -B_t^- q_t^-$, which shows that $w_t^- > w_t^+ > 0$ is not optimal, since the net cost of buying w_t^- and selling w_t^+ is higher than the cost of selling $q_t^- = w_t^+ - w_t^-$. In case iii), $w_t^- = w_t^+ = w_t$ implies $w_t(B_t^+ - B_t^-)$. As $B_t^- > B_t^+$, $w_t = 0$ minimizes the loss. Since w_t^- and w_t^+ cannot be both positive in any case, either $w_t^- = 0$ or $w_t^+ = 0$, $\forall\, t \in \mathcal{T}$ in any optimal solution. □

4.8.2 Appendix B: Proof of proposition 2

Proof. From (4.15) and (4.25),

$$r_{it} \leq \min\{O_{it},\, \bar{R}_{it}\}\,, \quad \forall\; i \in \mathcal{I}, t \in \mathcal{T}. \tag{4.33}$$

From (4.16),

$$\begin{aligned}\sum_{k\in\mathcal{K}(i,t)} kz_{itk} &= \bar{G}_{it} - r_{it},\ \forall\, i \in \mathcal{I}, t \in \mathcal{T},\\ &\geq\ \bar{G}_{it} - \max\{r_{it}\},\ \forall\, i \in \mathcal{I}, t \in \mathcal{T},\\ &=\ \bar{G}_{it} - \min\{O_{it},\ \bar{R}_{it}\},\ \forall\, i \in \mathcal{I}, t \in \mathcal{T}, && \text{(by Eq. 4.33)}\\ &=\ \max\{\bar{G}_{it} - O_{it},\ \bar{G}_{it} - \bar{R}_{it}\},\ \forall\, i \in \mathcal{I}, t \in \mathcal{T},\\ &\triangleq \underline{K}_{it}.\end{aligned}$$

Then, by (4.11) and (4.16), $k \geq\ \underline{K}_{it},\ \forall\ k \in \mathcal{K}(i,t)$. Similarly, from constraints (4.16),

$$\begin{aligned}\sum_{k\in\mathcal{K}(i,t)} kz_{itk} &= \bar{G}_{it} - r_{it},\ \forall\, i \in \mathcal{I}, t \in \mathcal{T},\\ &\leq\ \bar{G}_{it} - \min\{r_{it}\},\ \forall\, i \in \mathcal{I}, t \in \mathcal{T},\\ &=\ \bar{G}_{it} - \underline{R}_{it},\ \forall\, i \in \mathcal{I}, t \in \mathcal{T},\\ &\triangleq \bar{K}_{it},\end{aligned}$$

which by (4.11) and (4.16) implies $k \leq\ \bar{K}_{it},\ \forall\ k \in \mathcal{K}(i,t)$. □

4.8.3 Appendix C: Proof of proposition 3

Proof. To simplify the notation, we drop the indices $(i', t') \in \mathcal{I} \times \mathcal{T}$ from $\bar{K}_{i't'}$, $\bar{R}_{i't'}$, $\underline{R}_{i't'}$, $r_{i't'}$, $\mathcal{K}(i', t')$ and $z_{i't'k}$. In any feasible solution to P_B, P_E, variables y_{mt} are binary by (4.18) and r is integer by (4.14). By condition i), all available $\bar{G}$ generators can be active, which implies $\bar{K} = \bar{G}$ according to (5.57). Condition i) also implies $r \geq 0$ by (4.25). On the other hand, by (4.16) and Condition ii), $r \leq 2 = \bar{R}$. Therefore, for the analysis of the linear system with (4.11) and (4.16), we consider three cases:

1. $r = 0$: By conditions i) and ii),

$$\mathcal{K} = \{\bar{G},\ \bar{G} - 1,\ \bar{G} - 2\}. \tag{4.34}$$

Then, the linear system (4.11) and (4.16) can be written in extensive form as

$$z_{\bar{G}} + z_{\bar{G}-1} + z_{\bar{G}-2} = 1, \tag{4.35}$$

$$\bar{G}z_{\bar{G}} + (\bar{G} - 1)z_{\bar{G}-1} + (\bar{G} - 2)z_{\bar{G}-2} = \bar{G} - r. \tag{4.36}$$

By (4.30), $r = 0$ implies $z_k = 0\ \forall\ k < \bar{G}$. Then, by (4.11) $z_{\bar{G}} = 1$. Therefore, the system (4.35)-(4.36) has a unique integer solution.

2. $r = 1$: By (4.14) and (4.29), $r = 1$ implies $z_{\bar{G}} = 0$. Then, the system (4.35)-(4.36) reduces to

$$z_{\bar{G}-1} + z_{\bar{G}-2} = 1, \tag{4.37}$$

$$(\bar{G} - 1)z_{\bar{G}-1} + (\bar{G} - 2)z_{\bar{G}-2} = \bar{G} - 1, \tag{4.38}$$

with a unique integer solution $z_{\bar{G}-1} = 1$, $z_{\bar{G}-2} = 0$.

3. $r = 2$: By (4.14) and (4.29), $r = 2$ implies $z_{\bar{G}} = 0$, and the resulting system of equations

$$z_{\bar{G}-1} + z_{\bar{G}-2} = 1, \tag{4.39}$$

$$(\bar{G} - 1)z_{\bar{G}-1} + (\bar{G} - 2)z_{\bar{G}-2} = \bar{G} - 2. \tag{4.40}$$

has a unique integer solution $z_{\bar{G}-1} = 0$ and $z_{\bar{G}-2} = 1$.

Therefore, in models P_B, P_E with equations (4.29) and (4.30) and conditions $i)-iii)$ satisfied for some $(i', t') \in \mathcal{I} \times \mathcal{T}$, the system (4.11) and (4.16) for (i', t') has a unique solution and this solution is integer even if the integrality condition on the $z_{i't'k}$ variables is relaxed for (i', t') and $\forall\ k \in \mathcal{K}(i', t')$. □

Acknowledgment

This research was funded by Rio Tinto and by the NSERC–Engage and MITACS-Accelerate programs.

CHAPTER 5 ARTICLE 2: STOCHASTIC HYDROPOWER GENERATOR MAINTENANCE SCHEDULING VIA BENDERS DECOMPOSITION

Authors: Jesus A. Rodríguez, Miguel F. Anjos, Pascal Côté, Guy Desaulniers
Submitted to: European Journal of Operational Research.

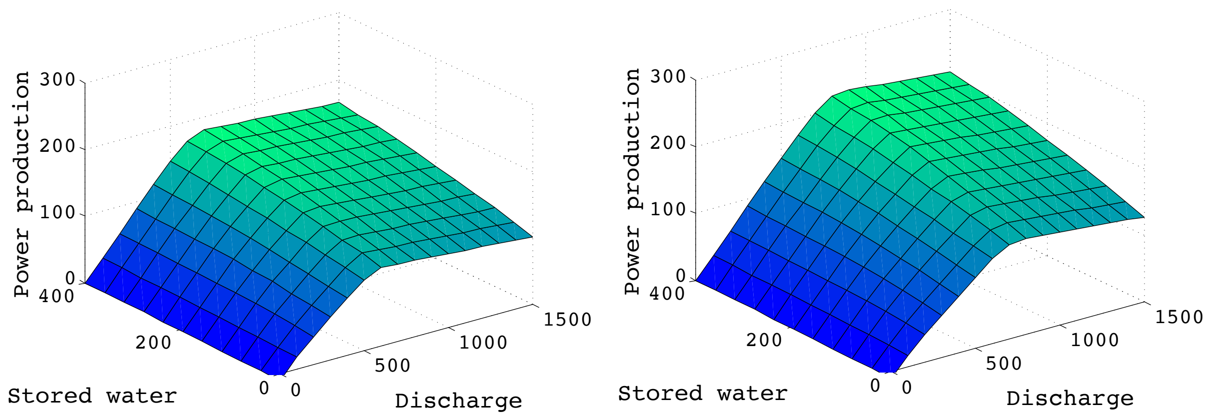
Abstract: Maintenance of power generators is essential for reliable and efficient electricity production. Because generators under maintenance are typically inactive, optimal planning of maintenance activities must consider the impact of maintenance outages on the system operation. However, finding a minimum cost maintenance schedule in hydropower systems is a challenging optimization problem due to the nonlinearity of the electricity production, the uncertainty of the water inflows and the intrinsic complexity of scheduling problems. We propose the first two-stage stochastic programming formulation for the hydropower generator maintenance scheduling problem, and we implement a parallelized Benders decomposition method with several acceleration techniques for its solution, considering a large number of scenarios. We apply statistical methods for selecting the best combination of acceleration techniques for the decomposition algorithm, and we compare the computational time of the parallelized decomposition against a mixed-integer linear programming solution approach using a testbed adapted from a real hydropower system in Canada.

Keywords: Stochastic programming, Benders decomposition, Parallel computing, Acceleration techniques, Hydropower maintenance scheduling.

5.1 Introduction

To guarantee the efficiency and reliability of electricity production, power producers carry out maintenance activities on a regular basis. As generators are usually inactive during maintenance, the economic impact of maintenance activities on the system operation must be considered. However, in hydropower systems this impact is difficult to estimate due to the nonlinearity of the hydroelectric generation, the uncertainty of the water inflows and the interdependence between multiple physical variables of the system.

A hydropower system is composed of powerhouses with turbine-generator units driven by the potential and kinetic energy of water. In each powerhouse, the hydroelectric generation is a function of the water level of the feeding reservoir or river, the discharged water through the turbines, the efficiency of the turbine-generator units and the energy loss due to the friction of the discharged water. If the turbine-generator units of a powerhouse have similar characteristics, the maximum power output p of the powerhouse with k active units can be represented by a function $p = f(s, u, k)$, where u is the water discharge and s is the stored water. We refer to this function as the Hydropower Production Function (HPF), whose nonlinearity is apparent in Fig 4.1. For short-term hydropower operation, the HPF has been



(a) Four generators (b) Five generators

Figure 5.1 Maximum power generation p in a powerhouse, for different values of u, s and k. Séguin et al. (2016) proposed a method for computing these surfaces.

represented with nonlinear functions (Arce, 2001), linear approximations (Conejo et al., 2002; Borghetti et al., 2008) and smoothing splines (Séguin et al., 2016), among others.

The hydropower operation must also take into consideration spatial and temporal interdependencies, since water discharges can feed downstream reservoirs, and current decisions determine future costs of the system, due to the effect of the water discharges on the stored water level. Furthermore, hydroelectric generation relies on water inflows from tributary rivers, snow-melt or rainfall which tend to be difficult to predict and can exhibit large variability. Scenario trees and scenario fans (Fig. 5.2) are some of the approaches used for representing the stochasticity of the water inflows (Séguin et al., 2017).

The generator maintenance scheduling problem (GMSP) consists in determining a calendar of maintenance outages with the best performance metric (such as reliability of the system, economic benefit or cost). In the GMSP, feasible schedules must satisfy constraints related to maintenance policies and resources as well as operational requirements, such as the minimum number of units available for operation. We address this problem in the context of hydropower systems, considering the aforementioned aspects, as well as the unique operating conditions of the hydroelectric generation, such as the uncertain water inflows and the nonlinearity of electricity production. We refer to this problem as the Stochastic Generator Maintenance

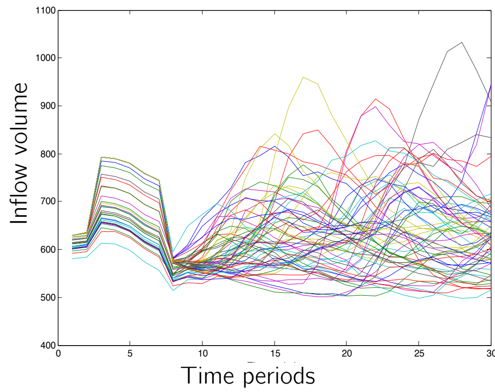


Figure 5.2 Scenario fan of water inflows. Each time series represents a scenario of forecasted water inflows.

Scheduling Problem (SGMSP) in hydropower systems. As the optimal scheduling of generator outages can increase the electricity production (Rodríguez et al., 2017), the impact of this problem is significant for hydropower producers.

Although the GMSP has been widely studied (Froger et al., 2016), the stochastic nature of the hydropower operation in combination with a realistic representation of the nonlinear hydroelectric generation has not yet been properly addressed. Feng et al. (2011) represented the power generation with fuzzy variables but omitted essential aspects of hydropower systems, such as the water storage levels and the uncertain water inflows. Foong et al. (2008) proposed an ant colony metaheuristic for maintenance scheduling with an oversimplified model of the hydropower operation. Kuzle et al. (2010) and Perez-Canto (2008) implemented a basic Benders decomposition method for the problem, without considering the nonlinearity of the electricity production and the stochastic water inflows. Recently, Rodríguez et al. (2017) proposed a mixed-integer linear programming (MILP) formulation for the deterministic GMSP in hydropower systems, with a convex approximation of the HPF. Rodríguez et al. (2017) showed that neglecting the nonlinearity of the HPF leads to significant overestimates of the electricity production and to suboptimal solutions in practice. As the resulting mathematical program is hard to solve in large instances of the problem, special solution methods that exploit its mathematical structure are necessary. Naturally, incorporating the water inflows uncertainty into the GMSP makes the problem even more challenging.

In this paper, we propose a two-stage stochastic optimization program for SGMSP in hydropower systems. This model is an extension of the deterministic MILP formulated by Rodríguez et al. (2017) with a linear approximation of the HPF. Using the Benders decomposition method, we partition the problem into a maintenance-only scheduling problem and scenario-wise operation subproblems. As the straightforward implementation of Benders decomposition is not a guarantee of efficient solution, we propose several enhancements to this method and we parallelize its execution. Using statistical methods, we select the best combination of the proposed acceleration techniques for the decomposition method, and we compare its performance against a MILP-based approach. For the tests, we consider a 4-powerhouse system with up to 200 inflow scenarios.

5.2 Mathematical programming models

In this section, we describe the optimization approach to the SGMSP, and we present its two-stage stochastic programming formulation.

5.2.1 Two-stage stochastic programming approach

In maintenance scheduling, the set of feasible maintenance decisions $\mathcal{Y}$ is defined by the maximum number of simultaneous outages, the time windows of maintenance activities and other relevant constraints. As the maintenance decisions $y \in \mathcal{Y}$ determine the set of available generators for electricity production in the planning horizon $\mathcal{T}$, we can compactly represent the GMSP as

$$\max_{y \in \mathcal{Y}} Q(y) - c^{\intercal} y, \tag{5.1}$$

where c is the cost vector of the maintenance activities, and $Q(y)$ is the operating profit during $\mathcal{T}$, corresponding to a maintenance schedule vector y. In hydropower systems, the water inflows uncertainty can be represented with a set of forecasted inflows (Fig. 5.2), which can be used to reformulate (5.1) as a two-stage stochastic program with maintenance scheduling decisions in the first stage and hydropower operation decisions for each water inflow scenario in the second stage (Fig. 5.3). As the actual scenario realization cannot be anticipated at the moment of determining the maintenance schedule y, we compute $Q(y)$ as the expected value of the profit over the set of scenarios Ω, with probability of occurrence φ_ω for scenario $\omega \in \Omega$, i.e.,

$$Q(y) = \sum_{\omega \in \Omega} \varphi_\omega Q_\omega(y),$$

where $Q_\omega(y)$ denotes the maximum cumulative operating profit corresponding to the maintenance schedule y, during $\mathcal{T}$, in scenario $\omega \in \Omega$, i.e.,

$$Q_\omega(y) = \max_{x_\omega \in \mathcal{X}(y, \xi_\omega)} \Theta(x_\omega). \tag{5.2}$$

The hydropower operation subproblem (5.2) determines the values of the operational vari-

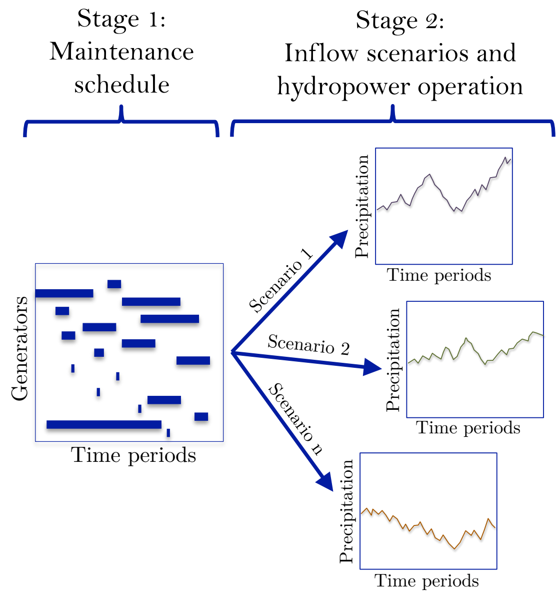


Figure 5.3 Generator maintenance scheduling as a two-stage stochastic problem. The maintenance schedule is defined in the first stage. Operating decisions take place in the second stage, once the inflows information is revealed

.

ables x_ω, such as water discharges and electricity production, that maximize the profit $\Theta(x_\omega)$ during $\mathcal{T}$. The feasible set $\mathcal{X}(y, \xi_\omega)$ of the decision variables is defined by the operational constraints of the problem, such as the water balance and the generation capacity, which depend on the maintenance schedule y and the water inflow parameters ξ_ω of the corresponding scenario. Naturally, problems (5.1) and (5.2) can be merged into a single deterministic equivalent mathematical program

$$\max_{\substack{y \in \mathcal{Y} \\ x_\omega \in \mathcal{X}(y, \xi_\omega)}} \sum_{\omega \in \Omega} \varphi_\omega \Theta(x_\omega) - c^\intercal y. \tag{5.3}$$

For a background on stochastic programming, we refer the reader to Birge and Louveaux (2011).

The next subsection presents the deterministic equivalent (5.3) of the two-stage stochastic program for the SGMSP in hydropower systems. This formulation is an extension of the model proposed by Rodríguez et al. (2017). Later we reformulate this problem for its solution via Benders decomposition.

5.2.2 Mathematical program

Consider a hydroelectric system with a set of powerhouses $\mathcal{I}$, and with a number of available generators $\bar{G}_{it}$ at each time period $t \in \mathcal{T}$ and powerhouse $i \in \mathcal{I}$. We assume that in each powerhouse the generators have similar characteristics. Let $\mathcal{M}$ be a list of generator maintenance activities to be completed within the planning horizon $\mathcal{T}$, with each activity requiring one generator outage. We define each maintenance activity m by i) the powerhouse where the activity must be executed, ii) the duration of the activity D_m, and iii) the time window $\mathcal{T}(m) \subseteq \mathcal{T}$ when the activity can initiate. Let $\mathcal{K}(i,t)$ be the set of numbers of generators that can be active at each time period and powerhouse. For determining the maintenance schedule, we define the binary variables $y_{mt} = 1$ if maintenance task $m \in \mathcal{M}$ starts at time period $t \in \mathcal{T}(m)$, 0 otherwise (5.4). We also define the binary variables $z_{itk} = 1$ if $k \in \mathcal{K}(i,t)$ generators are active in powerhouse $i \in \mathcal{I}$ at time period $t \in \mathcal{T}$, 0 otherwise (5.5).

$$y_{mt} \in \{0,1\},\ \forall\,(m,t) \in \mathcal{M} \times \mathcal{T}(m), \tag{5.4}$$

$$z_{itk} \in \{0,1\},\ \forall\,(i,t,k) \in \mathcal{I} \times \mathcal{T} \times \mathcal{K}(i,t). \tag{5.5}$$

For the SGMSP, we also define the following constraints that involve only first-stage maintenance decision variables:

$$\sum_{t \in \mathcal{T}(m)} y_{mt} = 1, \quad \forall\, m \in \mathcal{M}, \tag{5.6}$$

$$\sum_{\substack{m \in \mathcal{M}(i) \\ t' \in \mathcal{T}(m) \cap [\,t - D_m + 1,\, t\,]}} y_{mt'} = r_{it}, \quad \forall\,(i,t) \in \mathcal{I} \times \mathcal{T}, \tag{5.7}$$

$$r_{it} + \sum_{k \in \mathcal{K}(i,t)} k z_{itk} = \bar{G}_{it}, \quad \forall\,(i,t) \in \mathcal{I} \times \mathcal{T}, \tag{5.8}$$

$$\sum_{k \in \mathcal{K}(i,t)} z_{itk} = 1, \quad \forall\,(i,t) \in \mathcal{I} \times \mathcal{T}, \tag{5.9}$$

$$0 \leq r_{it} \leq O_{it}, \quad \forall\,(i,t) \in \mathcal{I} \times \mathcal{T}. \tag{5.10}$$

Constraints (5.6) enforce the completion of the set of maintenance activities $\mathcal{M}$ in the planning horizon $\mathcal{T}$. Constraints (5.7) compute the number of maintenance outages r_{it} at each time period and powerhouse. In (5.7) the value of r_{it} is determined by summing the variables $y_{mt'}$ corresponding to the set of activities $\mathcal{M}(i)$ in powerhouse i that could have started at time $t' \in \mathcal{T}(m)$ and still be in execution at time $t \in \mathcal{T}$ for having started in the interval $[\,t - D_m + 1,\, t\,]$.

Constraints (5.8) map the number of maintenance outages r_{it} into the binary variables z_{itk} that represent the number of active generators k at time period t and powerhouse i. By (5.9) and (5.5), only one z_{itk} variable is equal to one for each powerhouse and time period. Constraints (5.10) define the non-negativity of r_{it} and limit it to the maximum number of outages O_{it} at each time period and each powerhouse.

In addition, for the hydropower operation problem the following constraints are defined for each water inflow scenario $\omega \in \Omega$ and time period $t \in \mathcal{T}$.

$$0 \leq v_{it\omega}, \quad \forall\ (i,t,\omega) \in \mathcal{I} \times \mathcal{T} \times \Omega, \tag{5.11}$$

$$0 \leq u_{it\omega} \leq \bar{U}_{it}, \quad \forall\ (i,t,\omega) \in \mathcal{I} \times \mathcal{T} \times \Omega, \tag{5.12}$$

$$\underline{S}_{it} \leq s_{it\omega} \leq \bar{S}_{it}, \quad \forall\ (i,t,\omega) \in \mathcal{I} \times \mathcal{T} \times \Omega, \tag{5.13}$$

$$0 \leq q^{+}_{t\omega} \leq \bar{W}^{+}_{t}, \quad \forall\ (t,\omega) \in \mathcal{T} \times \Omega, \tag{5.14}$$

$$0 \leq q^{-}_{t\omega} \leq \bar{W}^{-}_{t}, \quad \forall\ (t,\omega) \in \mathcal{T} \times \Omega, \tag{5.15}$$

$$s_{it\omega} - s_{i(t-1)\omega} = \left(\xi_{it\omega} + \sum_{g \in \mathcal{U}(i)} (u_{gt\omega} + v_{gt\omega}) - u_{it\omega} - v_{it\omega} \right) F, \quad \forall\ (i,t,\omega) \in \mathcal{I} \times \mathcal{T} \times \Omega, \tag{5.16}$$

$$p_{itk\omega} \leq \beta^{0}_{h} + \beta^{u}_{h} u_{it\omega} + \beta^{s}_{h} s_{it\omega}, \quad \forall\, (i,t,k,h,\omega) \in \mathcal{I} \times \mathcal{T} \times \mathcal{K}(i,t) \times \mathcal{H}(i,k) \times \Omega, \tag{5.17}$$

$$0 \leq p_{itk\omega} \leq z_{itk} \bar{P}_{ik}, \quad \forall\, (i,t,k,\omega) \in \mathcal{I} \times \mathcal{T} \times \mathcal{K}(i,t) \times \Omega, \tag{5.18}$$

$$\sum_{k \in \mathcal{K}(i,t)} p_{itk\omega} = p_{it\omega}, \quad \forall\ (i,t,\omega) \in \mathcal{I} \times \mathcal{T} \times \Omega, \tag{5.19}$$

$$\sum_{i \in \mathcal{I}} p_{it\omega} + q^{-}_{t\omega} = A_t + q^{+}_{t\omega}, \quad \forall\ (t,\omega) \in \mathcal{T} \times \Omega. \tag{5.20}$$

Constraints (5.11)-(5.15) specify the bounds of the hydropower operation decision variables: water spill $v_{it\omega}$, water discharge $u_{it\omega}$, stored water in reservoirs $s_{it\omega}$, electricity purchase $q^{-}_{t\omega}$ and electricity sale $q^{+}_{t\omega}$, respectively. Constraints (5.16) ensure the mass balance at each time period $t \in \mathcal{T}$ and reservoir $i \in \mathcal{I}$, considering the inflows from upstream reservoirs $g \in \mathcal{U}(i)$, as well as the uncertain water inflows $\xi_{it\omega}$ of the respective scenario $\omega \in \Omega$. In (5.16), F is a scalar that converts the flow units (typically m^3/s) to the suitable units for its left hand side term, $s_{it\omega} - s_{i(t-1)\omega}$, i.e., the difference in stored water between consecutive periods (such as hm^3/day). Also in (5.16), the consistency with the initial stored water is ensured by defining $s_{i(t-1)} = S_{i0}$ for $t = 1$.

In (5.17), for given values of water discharge $u_{it\omega}$ and stored water level $s_{it\omega}$, the set of hyperplanes $\mathcal{H}(i,k)$ with parameters β^{0}_{h}, β^{u}_{h} and β^{s}_{h}, approximates the power production $p_{itk\omega}$ corresponding to $k \in \mathcal{K}(i,t)$ active generators in powerhouse $i \in \mathcal{I}$. Constraints (5.18) restrict the generation capacity according to the number k of active generators, which is indicated by the binary variable z_{itk}. Thus, when the number of active generators is not equal to $\bar{k}$ ($z_{it\bar{k}} = 0$), the power production for this number of generators is set to zero ($p_{it\bar{k}\omega} = 0$). Constraints (5.19) compute the power generation $p_{it\omega}$ in each powerhouse, time period and scenario by summing the power production $p_{itk\omega}$ over the set of numbers of active generators $\mathcal{K}(i,t)$.

At each time period and scenario, the power balance is enforced by (5.20). In this balance, the total power injections into the system equal the power withdrawals. The injections correspond to the sum of the hydroelectric generation $p_{it\omega}$ and the electricity purchase $q^-_{t\omega}$. The power withdrawals are the electricity load A_t and the electricity sales $q^+_{t\omega}$.
Finally, the objective function of the complete problem is the sum of the expected profit of the electricity trade minus the costs of maintenance activities,

$$\underset{\substack{q^+,q^-,u,v,s,\\ r,p,y,z}}{\text{maximize}} \quad \sum_{\substack{t\in\mathcal{T}\\ \omega\in\Omega}} \varphi_\omega \Big(B^+_t q^+_{t\omega} - B^-_t q^-_{t\omega}\Big) - \sum_{\substack{m\in\mathcal{M}\\ t\in\mathcal{T}(m)}} C_{mt} y_{mt}, \tag{5.21}$$

where C_{mt} is the cost of maintenance activity m starting at time t, and B^-_t, B^+_t are the electricity prices of purchase and sale, respectively, at period t. Therefore, the two-stage stochastic program for the SGMSP is

$$\text{maximize (5.21) subject to (5.4)} - \text{(5.20)}. \tag{SGMSP}$$

To reduce the number of variables in (5.5) and the number of constraints in (5.17), (5.18) we define the set $\mathcal{K}(i,t)$ using the time windows of the maintenance activities, as proposed in Rodríguez et al. (2017) (see Appendix A.1 in Section 5.8.1).

5.3 Solution strategy

Because hydrological predictions are typically subject to uncertainty, the forecasted inflows can exhibit large differences (Fig. 5.2), so solving the SGMSP with a small number of scenarios can significantly reduce the quality of the information, which results in suboptimal decisions in practice. Therefore, a sufficiently large number of representative scenarios should be included into the model, in order to find solutions with the best average performance on the wide spectrum of inflows. However, as this number of scenarios can lead to a very large problem, we use Benders decomposition for its solution.

5.3.1 The Benders decomposition method

Benders decomposition (Benders, 1962) is a solution procedure based on the idea of partitioning a mathematical program into a relaxed master problem and a convex subproblem. The decomposition algorithm solves the master problem, fixes its solution into the subproblem, solves the subproblem and uses its dual information to generate cuts that approximate the cost function or the feasible space of the subproblem into the master problem. For a formal presentation of this method, consider the mathematical program

$$\begin{aligned} \underset{x,y}{\text{maximize}}\ & c^\intercal x + f(y) \\ \text{s.t.}\ & Ax + F(y) \leq b, \\ & x \geq 0, \\ & y \in S, \end{aligned} \tag{P}$$

where S is a posssibly nonconvex feasible set. In this problem, y and x are vectors of decision variables, $c \in \mathbb{R}^n$ and $b \in \mathbb{R}^m$ are constant vectors, $A \in \mathbb{R}^{n \times m}$ is a constant matrix, and $F(y)$, $f(y)$ are, respectively, m-component and scalar functions on y. By fixing the so-called complicating variables $\bar{y} \in S$, the resulting subproblem

$$\begin{aligned} Q(y) = \underset{x}{\text{maximize}}\ & c^\intercal x \\ \text{s.t.}\ & Ax \leq b - F(y) \\ & x \geq 0, \end{aligned} \tag{SP}$$

is convex, and thus much easier to solve. Benders (1962) showed that with the extreme dual solutions of SP, the original problem P can be rewritten as

$$\begin{aligned} \underset{z^{SP},y}{\text{maximize}}\ & z^{SP} + f(y) \\ \text{s.t.}\ & z^{SP} \leq [b - F(y)]^\intercal \pi^p,\ \forall\, p \in \mathcal{P}, \\ & y \in S, \end{aligned} \tag{MP}$$

which is the Benders Master Problem (MP). In this problem, $\mathcal{P}$ is the set of extreme solutions, π^p is the dual solution of SP corresponding to the extreme point $p \in \mathcal{P}$ and z^{SP} is the minimum value of the dual problem. In MP the constraints

$$z^{SP} \leq [b - F(y)]^\intercal \pi^p,\ \forall\, p \in \mathcal{P}, \tag{5.22}$$

are referred to as optimality cuts as they remove non-optimal solutions from the master problem. In cases where the master problem solution can result in an infeasible subproblem, feasibility cuts can also be included to remove master problem solutions that are infeasible for the complete problem. These cuts can be computed from the extreme rays of the dual subproblem.

Given that the set of extreme dual solutions $\mathcal{P}$ is potentially large, the Benders decomposition method relaxes the master problem by including only a subset $\mathcal{P}^R \subset \mathcal{P}$ of extreme solutions, which is empty in the initialization of the algorithm. We refer to this problem as the Relaxed Master Problem (RMP). At each iteration, the algorithm solves the RMP, fixes its solution into the subproblem, and solves the subproblem to obtain a new dual extreme point π^p that corresponds to a violated optimality cut. This cut is then included into the RMP for the next iteration. The procedure continues until reaching a specified gap between the upper bound UB^P and the lower bound LB^P on the optimal value Z^{P*} of the complete problem P. The upper bound UB^P is the optimal value of the RMP at the current iteration, and the lower bound LB^P is the objective value of the incumbent solution. At each iteration j, the Benders algorithm computes the objective value $Z^P(\bar{y}_j)$ of the complete problem, corresponding to the master problem solution $\bar{y}_j$, using the optimal value $Q(\bar{y}_j)$ of the subproblem, i.e.,

$$Z^P(\bar{y}_j) = Q(\bar{y}_j) + f(\bar{y}_j). \tag{5.23}$$

Then, the lower bound on Z^{P*} at the Jth iteration is

$$LB^P = \max_{0 \leq j \leq J} \left\{ Z^P(\bar{y}_j) \right\}. \tag{5.24}$$

5.3.2 Benders reformulation of the SGMSP

To implement the decomposition algorithm, we derive the subproblem SP and the relaxed master problem RMP for the SGMSP. In this reformulation, SP is the hydropower production problem, and RMP is the maintenance scheduling problem with the binary variables defined in (5.4), (5.5), which we compactly denote y, z. According to this partitioning of the problem, (5.18) are the linking constraints, i.e., the subproblem constraints where the decision variables of the master problem are fixed.

Subproblem

Given a master problem solution $(\bar{y}, \bar{z})$, we set $z = \bar{z}$ in (5.18) to obtain the per scenario $\omega \in \Omega$ subproblems, which consist in maximizing the profit of the electricity production, subject to the operational constraints (5.11)-(5.20), i.e.,

$$Q_\omega(\bar{z}) = \underset{q^+,q^-,u,v,s}{\text{maximize}} \quad \sum_{t\in\mathcal{T}}(B_t^+ q_{t\omega}^+ - B_t^- q_{t\omega}^-) \tag{5.25}$$

subject to

$$s_{it\omega} - s_{i(t-1)\omega} + F\Bigg(u_{it\omega} + v_{it\omega} - \sum_{g\in\mathcal{U}(i)} (u_{gt\omega} + v_{gt\omega})\Bigg) = F\xi_{it\omega} \perp \pi_{it\omega}, \quad \forall\, (i,t) \in \mathcal{I}\times\mathcal{T}, \tag{5.26}$$

$$p_{itk\omega} - \beta_h^u u_{it\omega} - \beta_h^s s_{it\omega} \le \beta_h^0 \perp \gamma_{itkh\omega}, \quad \forall\, (i,t,k,h) \in \mathcal{I}\times\mathcal{T}\times\mathcal{K}(i,t)\times\mathcal{H}(i,k), \tag{5.27}$$

$$0 \le p_{itk\omega} \le \bar{z}_{itk}\bar{P}_{ik} \perp \lambda_{itk\omega}, \quad \forall\, (i,t,k) \in \mathcal{I}\times\mathcal{T}\times\mathcal{K}(i,t), \tag{5.28}$$

$$\sum_{i\in I} p_{it\omega} + q_{t\omega}^- - q_{t\omega}^+ = A_t \perp \psi_{t\omega}, \quad \forall\, t\in\mathcal{T}, \tag{5.29}$$

$$\sum_{k\in\mathcal{K}(i,t)} p_{itk\omega} - p_{it\omega} = 0 \perp \theta_{it\omega}, \quad \forall\, (i,t) \in \mathcal{I}\times\mathcal{T}, \tag{5.30}$$

$$0 \le v_{it}, \qquad \forall\, (i,t) \in \mathcal{I}\times\mathcal{T}, \tag{5.31}$$

$$0 \le u_{it\omega} \le \bar{U}_{it} \quad (\alpha_{it\omega}^u), \quad \forall\, (i,t) \in \mathcal{I}\times\mathcal{T}, \tag{5.32}$$

$$\underline{S}_{it} \le s_{it\omega} \le \bar{S}_{it} \quad (\alpha_{it\omega}^s), \quad \forall\, (i,t) \in \mathcal{I}\times\mathcal{T}, \tag{5.33}$$

$$0 \le q_{t\omega}^+ \le \bar{W}_t^+ \quad (\alpha_{t\omega}^+), \quad \forall\, t\in\mathcal{T}, \tag{5.34}$$

$$0 \le q_{t\omega}^- \le \bar{W}_t^- \quad (\alpha_{t\omega}^-), \quad \forall\, t\in\mathcal{T}, \tag{5.35}$$

where $\pi_{it\omega}$, $\gamma_{itkh\omega}$, $\lambda_{itk\omega}$, $\psi_{t\omega}$ and $\theta_{it\omega}$ denote the dual variables of constraints (5.26)-(5.30), respectively, and the symbol $\perp$ indicates the complementarity of the constraint with the corresponding dual variable.

In order to reduce the subproblem size we specify (5.32)-(5.35) as variable bounds, so that they can be treated implicitly by the linear programming (LP) solver through the bounded variable simplex method. Because (5.32)-(5.35) are not specified as general constraints, their dual variables are not explicitly defined. For each bound constraint, we denote by α (in parentheses) its dual variable which is equal to the reduced cost of the corresponding variable.

Master problem

The Benders Master Problem (BMP) for the SGMSP maximizes the expected profit of the electricity production z^{SP} minus the maintenance cost, subject to the optimality cuts and the constraints of the original problem that involve only the maintenance decisions. Thus, the BMP is

$$\underset{y,z,z^{SP}}{\text{maximize}} \quad z^{SP} - \sum_{\substack{m \in \mathcal{M},\\ t \in \mathcal{T}(m)}} C_{mt} y_{mt} \tag{5.36}$$

subject to

$$\text{Eqs. } (5.4) - (5.10),$$

$$z^{SP} \leq \sum_{\omega \in \Omega} \varphi_\omega b_{\omega p}, \ \forall \ p \in \mathcal{P}^R, \tag{5.37}$$

$$z^{SP} \leq UB^{SP}, \tag{5.38}$$

where (5.37) are the optimality cuts corresponding to a subset $\mathcal{P}^R \subset \mathcal{P}$ of extreme solutions, and $b_{\omega p}$ is the cut term corresponding to solution $p \in \mathcal{P}^R$, in scenario $\omega \in \Omega$. At each iteration, a new solution is explored and hence the number of optimality cuts increases, unless the decomposition algorithm includes a cut removal procedure. As the Benders algorithm starts without optimality cuts, (5.38) prevents the unboundedness of the master problem at the first iteration. This constraint defines an initial upper bound UB^{SP} of the subproblem optimal value z^{SP}. Section 5.4.1 presents a method for computing tight values of UB^{SP}. In this master problem no feasibility cuts are necessary, due to the assumptions in Appendix A.2 (see Section 5.8.1). The computation of (5.37) is described next.

Optimality cuts

As shown in Section 5.3.1, the optimality cuts are calculated from the subproblem's dual solutions. Due to the definition of (5.32)-(5.35) as variable bounds, their dual variables are not explicitly available. Instead, for these bounds we use the reduced costs of the corresponding BSP primal variables to calculate their dual contribution. Thus, we compute the cut term $b_{\omega p}$ in (5.37) as

$$b_{\omega p} = b^1_{\omega p} + b^2_{\omega p}, \ \forall \, (\omega, p) \in \Omega \times \mathcal{P}, \tag{5.39}$$

where $b^1_{\omega p}$ is the dual contribution of (5.26)-(5.29), and $b^2_{\omega p}$ is the dual contribution of (5.32)-(5.35). For a given extreme solution $p \in \mathcal{P}$, we calculate $b^1_{\omega p}$ as the sum of the products between the right-hand side terms of (5.26)-(5.29), and the corresponding dual variables $\pi^p_{it\omega}$, $\gamma^p_{itkh\omega}$, $\lambda^p_{itk\omega}$, $\psi^p_{t\omega}$, i.e.,

$$
\begin{aligned}
b^1_{\omega p} = \sum_{t \in \mathcal{T}} \Bigg(A_t \psi^p_{t\omega} + \sum_{i \in \mathcal{I}} \Bigg(F \xi_{it\omega} \pi^p_{it\omega} \\
+ \sum_{k \in \mathcal{K}(i,t)} \Bigg(z_{itk} \bar{P}_{ik} \lambda^p_{itk\omega} + \sum_{h \in \mathcal{H}(i,k)} \beta^0_h \gamma^p_{itkh\omega} \Bigg)\Bigg)\Bigg), \forall (\omega, p) \in \Omega \times \mathcal{P}.
\end{aligned} \tag{5.40}
$$

Notice that in (5.40), we discarded the terms corresponding to constraints (5.30) because their right-hand side is 0.

For $b^2_{\omega p}$, we multiply each bound by the value of the corresponding dual variable $\alpha^{pu}_{it\omega}$, $\alpha^{ps}_{it\omega}$, $\alpha^{p+}_{t\omega}$, $\alpha^{p-}_{t\omega}$ in the solution $p \in \mathcal{P}$. That is,

$$
\begin{aligned}
b^2_{\omega p} = \sum_{t \in \mathcal{T}} \Bigg(\bar{W}^-_t \alpha^{p-}_{t\omega} + \bar{W}^+_t \alpha^{p+}_{t\omega} + \sum_{i \in \mathcal{I}} \Bigg(\bar{U}_{it} \alpha^{pu}_{it\omega} \\
+ \bar{S}_{it} \alpha^{ps}_{it\omega} [\alpha^{ps}_{it\omega} > 0] + \underline{S}_{it} \alpha^{ps}_{it\omega} [\alpha^{ps}_{it\omega} < 0] \Bigg)\Bigg), \ \forall (\omega, p) \in \Omega \times \mathcal{P}.
\end{aligned} \tag{5.41}
$$

Since the water discharge $s_{it\omega}$ has a lower bound $\underline{S}_{it}$, for the computation of $b^2_{\omega p}$ we sum either $\bar{S}_{it} \alpha^{ps}_{it\omega}$ or $\underline{S}_{it} \alpha^{ps}_{it\omega}$, depending on the sign of the corresponding dual value $\alpha^{ps}_{it\omega}$, as indicated by the Iverson brackets in (5.41). A positive dual value means that the upper bound is active, whereas a negative one indicates that the lower bound is binding.

5.4 Acceleration techniques for Benders decomposition

Although the divide and conquer principle of decomposition methods is a promising idea to reduce the computational effort, a straightforward implementation of the Benders algorithm can perform poorly due to the number of iterations required to converge, the time per iteration, and the growing size of the master problem as a result of the cuts that are included at each iteration. In response to these challenges, several ideas have been proposed to accelerate the Benders decomposition method, such as:

- Use a formulation with a tight continuous relaxation. The stronger the formulation, the faster the convergence (Magnanti and Wong, 1981).
- When the dual subproblem has multiple solutions, select the extreme point that produces the strongest cut (Magnanti and Wong, 1981; Papadakos, 2008).
- Solve a relaxed or partially relaxed master problem in the initial iterations. The cuts

obtained from these solutions are also valid for the integer master problem (Cordeau et al., 2001).

— In the master problem, include constraints and variables that help to approximate the original problem (Santoso et al., 2005; Crainic et al., 2016; Gendron et al., 2016).

— Solve the master problem in a branch and cut approach, with Benders cuts generated each time that a feasible integer node is found in the branching tree of the master problem (Fortz and Poss, 2009; Gendron et al., 2016; Fischetti et al., 2016a,b).

— To reduce the oscillation of the subproblem solution, use a trust region approach or a stabilization method (Santoso et al., 2005; Fischetti et al., 2016a).

— Besides the Benders cuts, generate additional cuts (combinatorial cuts, knapsack cuts, among others) from the explored master problem solutions (Santoso et al., 2005; Fischetti et al., 2016a; Gendron et al., 2016; Ljubić et al., 2017).

For a recent review on Benders decomposition, we refer the reader to Rahmaniani et al. (2017).

Furthermore, as the subproblems can be solved independently once the master problem solution is fixed, parallelization of the scenario-wise subproblems is a natural alternative for speeding up the Benders algorithm. Nevertheless, an efficient parallel computing implementation must consider particular aspects, such as the parallelization protocol and the fine-grained design of the parallel algorithm, in order to reduce the communication overhead, to improve the load balance and to exploit the intrinsic parallelism of the solution method. Previous works have addressed some of these aspects in the context of stochastic programming (e.g. Nielsen and Zenios, 1997; Linderoth and Wright, 2003).

5.4.1 Implemented techniques

For speeding up the Benders decomposition method, we tested the following strategies, as discussed afterwards: 1) valid inequalities (Rodríguez et al., 2017), 2) warm start, 3) multi-phase relaxation (Cordeau et al., 2001), 4) special ordered sets (Beale and Tomlin, 1970), 5) combinatorial cuts (Codato and Fischetti, 2006), 6) presolve, 7) integer rounding cuts (Santoso et al., 2005), 8) parallelization.

Valid inequalities (VI)

As tight formulations can be favorable for Benders decomposition (Magnanti and Wong, 1981), we test the effect of the valid inequalities (5.42)-(5.43) (Rodríguez et al., 2017) and

(5.44) on the performance of the decomposition method for the SGMSP.

$$\sum_{\substack{m \in \mathcal{M}(i) \\ t' \in \mathcal{T}(m) \cap [t - D_m + 1, t]}} y_{mt'} + z_{itk} \leq 1 \tag{5.42}$$

$$\text{for } k = \bar{G}_{it}, \forall \ (i, m, t) \in \mathcal{I} \times \mathcal{M}(i) \times \mathcal{T},$$

$$\sum_{k \in \mathcal{K}(i,t) \setminus \{\bar{G}_{it}\}} z_{itk} \leq r_{it}, \ \forall \ (i, t) \in \mathcal{I} \times \mathcal{T}, \tag{5.43}$$

$$r_{it} + \sum_{k \in \mathcal{K}(i,t) \setminus \{\underline{K}_{it}\}} (k - \underline{K}_{it}) z_{itk} \leq \bar{R}_{it}, \ \forall \ (i, t) \in \mathcal{I}, \times \mathcal{T}, \tag{5.44}$$

where $\underline{K}_{it}$ and $\bar{R}_{it}$ are respectively the minimum number of active generators and the maximum number of activities simultaneously in execution at (i, t). For a derivation of (5.42)-(5.44), see Appendix A.3 in Section 5.8.1.

Warm start (WS)

In a branch and bound process, the objective value of the current best feasible solution cuts off sections of the branching tree with no potential of harboring an optimal solution. The tighter the cutoff value, the fewer the number of nodes to be explored in the tree. In MILP solvers, cutoff values can be user-defined or can be computed from user-supplied initial solutions. Even if the initial solution is infeasible, MILP solvers can apply re-optimization or heuristics to obtain a new feasible solution and a corresponding cutoff value (FICO, 2017). At any iteration of the Benders algorithm, the lower bound LB^P in (5.24) is naturally a cutoff value for the master problem. Therefore, at each iteration we specify to the solver a cutoff value $LB^P - \epsilon$, where $\epsilon = TOL \cdot |LB^P|$, and TOL is the default relative optimality tolerance of the MILP solver. In addition, we provide the master problem solution of the previous iteration as an initial solution to the MILP solver for the new iteration.
As tightening bounds of variables can also make the search more efficient, at the first step of the algorithm we obtain an initial upper bound UB^{SP} of z^{SP} in (5.38), computed as the optimal value of the linear relaxation of the complete problem plus the maximum maintenance cost of the LP solution of the initial master problem (i.e., with no optimality cuts and with fixed $z^{SP} = 0$). Moreover, at each iteration we define the current solution value of z^{SP} in the master problem as the upper bound UB^{SP} for the next iteration.

Multi-phase relaxation (MR)

Considering that the solutions to a RMP can generate valid cuts (Cordeau et al., 2001), we evaluate the effect of several relaxation schemes. For the master problem (5.4)-(5.10), (5.36)-(5.38), we define four relaxation levels of the binary variables y, z (Table 5.1). Among the possible sequences for applying these relaxations, we consider those that start with a complete linear relaxation (relaxation level 3) and in the subsequent phases solve an integer or partially integer RMP (relaxation levels 0, 1 or 2). To ensure a feasible solution, the last phase solves the integer master problem. We compare these relaxation sequences against a standard single-phase algorithm (without a relaxation phase, defined as sequence 0 in Table 5.2).

Table 5.1 Configuration of relaxation levels

Relaxation level	Binary variables	Linear relaxation type
0	y, z	No relaxation
1	y	Partial
2	z	Partial
3	-	Complete

Table 5.2 Sequences of relaxation levels for multi-phase relaxation

Index sequence	Relaxation sequence
0	0
1	3, 0
2	3, 2, 0
3	3, 1, 0
4	3, 1, 2, 0
5	3, 2, 1, 0

The relaxation of the master problem at the initial stages helps to quickly generate optimality cuts. Nevertheless, to prevent an excessive number of cuts that can slow down the decomposition algorithm, each relaxation stage can be finished when certain conditions are met, such as the maximum number of cuts at the stage or the minimum optimality gap of the stage.

Special ordered sets (SOS)

In a branch and bound algorithm, branching on sets of variables, instead of individual variables, can reduce the computational time. Special Ordered Sets (SOS) allow specifying sets of variables for branching decisions (Beale and Tomlin, 1970). A set of variables ordered by a reference value, and with at most n consecutive non-zero variables in the set, can be specified as a SOS of type n (SOS-n), where $n \leq 2$. When branching on a SOS-1, a position in the ordered set is chosen, and all variables above and below the chosen position are forced to a zero value (Beale and Tomlin, 1970).
In the master problem (5.4)-(5.10), (5.36)-(5.38), the variables z_{itk} form a set ordered by k, for each time period t and powerhouse i. Thus, we replace the binary condition on z_{itk} (5.5) with the following SOS-1 definition

$$\text{SOS-1}_{it} = \{z_{itk} \to k \,:\, k \in \mathcal{K}(i,t)\} \;\forall\; (i,t) \in \{\,\mathcal{I} \times \mathcal{T} : |\,\mathcal{K}(i,t)\,| > 2\},$$

where the arrow symbol $\to$ indicates that k is the ordering value of the set. Since SOS work better when the cardinality of the set is not very small (FICO, 2017), we define a SOS-1_{it} only when the size of the set is greater than 2.
Moreover, when $B_t^- \geq B_t^+$ and $A_t \leq \bar{P}_{ik} \leq W_t$, the order of the variables z_{itk} in the master problem can be enforced by the constraint,

$$z^{SP} \leq \sum_{t \in \mathcal{T}} B_t^+ \Bigg(\sum_{\substack{i \in \mathcal{I}, \\ k \in \mathcal{K}(i,t)}} \bar{P}_{ik} z_{itk} - A_t \Bigg), \tag{5.45}$$

which defines an upper bound of the subproblem objective value (5.25). In (5.45), the order of the variables z_{itk} for each set (i,t) is determined by the generation capacity $\bar{P}_{ik}$, which increases with the number of generators k. Since $B_t^- \geq B_t^+$, buying electricity for selling it (i.e., electricity arbitrage) is suboptimal, so in an optimal solution, only the surplus electricity production can be sold. For a given number k of active generators, the maximum surplus electricity is the capacity $\bar{P}_{ik}$ minus the load A_t (5.29). When the assumption $A_t \leq \bar{P}_{ik} \leq W_t$ does not hold, (5.45) must be replaced by the inequality in Appendix A.4 (see Section 5.8.1).

Combinatorial cuts (CC)

Combinatorial Benders cuts (CBC) (Codato and Fischetti, 2006) have been proposed to remove infeasible solutions in mathematical programs with binary variables. In contrast with the traditional Benders feasibility cuts, which are computed from the subproblem dual extreme rays, CBC exclude the current binary solution $\bar{x}$ by forcing a change of value in at least one variable of $\bar{x}$. Given the variables x_j with index set $\mathcal{J}$, CBC are defined as

$$\sum_{j \in \mathcal{S}} (1 - x_j) + \sum_{j \notin \mathcal{S}} x_j \geq 1, \tag{5.46}$$

where $\mathcal{S}$ is the set of variables in $\bar{x}$ with value 1, i.e., $\mathcal{S} = \{j \in \mathcal{J} : \bar{x}_j = 1\}$, and its complement is $\mathcal{S}' = \{j \in \mathcal{J} : \bar{x}_j = 0\}$. We obtain a stronger inequality than (5.46), by forcing at least one variable in each set, $\mathcal{S}$ and $\mathcal{S}'$, to have a different value, i.e,

$$\sum_{j \in \mathcal{S}} x_j \leq |\mathcal{S}| - 1, \tag{5.47}$$

$$\sum_{j \notin \mathcal{S}} x_j \geq 1 \tag{5.48}$$

Then, from (5.47) and (5.48), we obtain

$$\sum_{j \in \mathcal{S}} x_j - \sum_{j \notin \mathcal{S}} x_j \leq |\mathcal{S}| - 2. \tag{5.49}$$

Proposition 4. *The combinatorial cut* (5.49) *dominates the standard CBC* (5.46).

Proof. Inequality (5.46) can be rewritten as

$$\sum_{j \in \mathcal{S}} x_j - \sum_{j \notin \mathcal{S}} x_j \leq |\mathcal{S}| - 1. \tag{5.50}$$

As (5.50) and (5.49) have equal left-hand side, and the right-hand side of (5.50) is greater than the right-hand side of (5.49), then (5.49) dominates (5.46). □

Applying (5.49) to cut a suboptimal solution $\bar{y}$ in (5.4)-(5.10), (5.36)-(5.38), gives

$$\sum_{(m,t) \in \mathcal{S}_y} y_{mt} - \sum_{(m,t) \notin \mathcal{S}_y} y_{mt} \leq |\mathcal{M}| - 2, \tag{5.51}$$

where $\mathcal{S}_y = \{(m,t) \in \mathcal{M} \times \mathcal{T}(m) : \bar{y}_{mt} = 1\}$. Notice that $|\mathcal{S}_y| = |\mathcal{M}|$, since for each activity m there is a variable $\bar{y}_{mt} = 1$ (5.6).

Furthermore, when the costs of the tasks are independent of the starting time, i.e., when $C_{mt} = C_m$, $\forall\,(m,t) \in \mathcal{M} \times \mathcal{T}(m)$, different solutions $\bar{y}$ that correspond to the same solution $\bar{z}$, would have the same objective value. In this case, a valid cut is

$$\sum_{(i,t,k) \in \mathcal{S}_z} z_{itk} - \sum_{(i,t,k) \notin \mathcal{S}_z} z_{itk} \leq |\mathcal{I}||\mathcal{T}| - 2. \tag{5.52}$$

In (5.52), $\mathcal{S}_z = \{(i,t,k) \in \mathcal{I} \times \mathcal{T} \times \mathcal{K}(i,t) : \bar{z}_{itk} = 1\}$, with cardinality $\mathcal{S}_z = |\mathcal{I}||\mathcal{T}|$, since by (5.9), for each time period t and powerhouse i, exactly one variable $\bar{z}_{itk}$ is equal to 1. To prevent removing optimal solutions, we only apply the cuts (5.52) and (5.51) when the objective value of the solution $(\bar{y}, \bar{z})$ is lower than the cutoff value, that is, when $Z^P(\bar{y}, \bar{z}) < LB^P - \epsilon$, where ϵ is as defined in Section 5.4.1.

Presolve (PS)

As presolve is a key element for efficiently solving MILP problems (Bixby et al., 1999), several MILP solvers presolve the problem before the branch and cut procedure (Bixby et al., 1999). A presolve routine reduces the problem through several operations such as tightening bounds, coefficient reduction, removal of redundant columns and rows, and fixing variables based on logical implications or dual information (FICO, 2017; Bixby et al., 1999). By reducing the domain of the variables and removing fractional solutions, presolve can improve the upper and the lower bound of MILP problems (Bixby et al., 1999). However, as in Benders decomposition only part of the original problem information is included into the RMP, the potential of presolving the RMP is reduced. Furthermore, as new rows are included at each iteration of the Benders algorithm, presolve operations such as reduced cost fixing can produce inconsistent solutions if applied to the RMP and fixed for subsequent iterations. In contrast, presolving the complete problem gives problem reductions that are valid for the RMP through all iterations. Therefore, we can accelerate the Benders algorithm with an initialization step that 1) applies to the complete problem (5.4)-(5.21) a presolve routine with all presolve operations activated, and 2) in the RMP fixes for all iterations of the Benders algorithm the binary variables that after presolving the complete problem are set to one of their bounds. Notice that the values of the variables fixed during presolve must be explicitly retrieved from the MILP solver because their values can be different from the linear relaxation solution.

Integer rounding cuts (IRC)

Let $c^\intercal$ be the coefficient vector of y in the master problem. Since the lower bound LB^P of the complete problem is also valid for the master problem, combining the bound $LB^P \leq c^\intercal y + Q^{SP}$, with the optimality cut $Q^{SP} \leq a^\intercal y + b$, gives the inequality $LB^P \leq (c+a)^\intercal y + b$, which can be tightened with integer rounding and division by the Greatest Common Divisor (GCD) of $\lceil c + a \rceil$ (Santoso et al., 2005; Chen et al., 2011). Thus,

$$\frac{\lceil LB^P - b \rceil}{\text{GCD}} \leq \left(\frac{\lceil c + a \rceil}{\text{GCD}} \right)^\intercal y, \tag{5.53}$$

is a valid cut for the master problem. As the bound LB^P increases as the algorithm progresses, an IRC (5.53) can become weak in subsequent iterations. In the tested instances of the SGMSP, we observed that keeping only the most recent IRC had a better impact on the computational time than keeping all the generated IRC (5.53) and updating their constant term when the bound LB^P improves.

Parallelization

For the parallelization of the Benders algorithm, we implemented a master-slave approach, where the slave processors solve the subproblem and compute the cut terms, and the master process includes the cuts, solves the master problem and controls the execution of the algorithm (Fig. 5.4). The master process runs on a computer server with a MILP solver, and the slave processes run independently on a computer cluster with an open source linear programming solver.

We used the Message Passing Interface (MPI) standard as a parallel programming protocol. Although MPI requires explicit instructions for communications among processes, some MPI implementations are portable, free and can use both shared and distributed memory. Furthermore, MPI incorporates routines for high-performance collective communication that are suitable for our master-slave implementation of the decomposition algorithm.

5.4.2 Implementation details

The code was written in C++ with the modeling libraries Xpress BCL. The master problem was solved with the MILP solver Xpress-MP, and the subproblems were solved with the open-source linear programming solver Clp. For the parallelization we used MPICH and the Intel MPI Library. In BCL, we specified the Benders optimality cuts as *delayed rows*. This cut definition is appropriate when most of the cuts are unlikely to be active, since only the violated cuts are reintroduced by the solver when a new solution is found. Other cuts that we proposed (valid inequalities, combinatorial cuts and integer rounding cuts) were defined in BCL as *model cuts*, since they can be included by the solver to remove fractional solutions, but are not necessary to obtain feasible solutions. Furthermore, to avoid a large number of combinatorial cuts and integer rounding cuts, we kept only the cuts generated in the previous iteration.

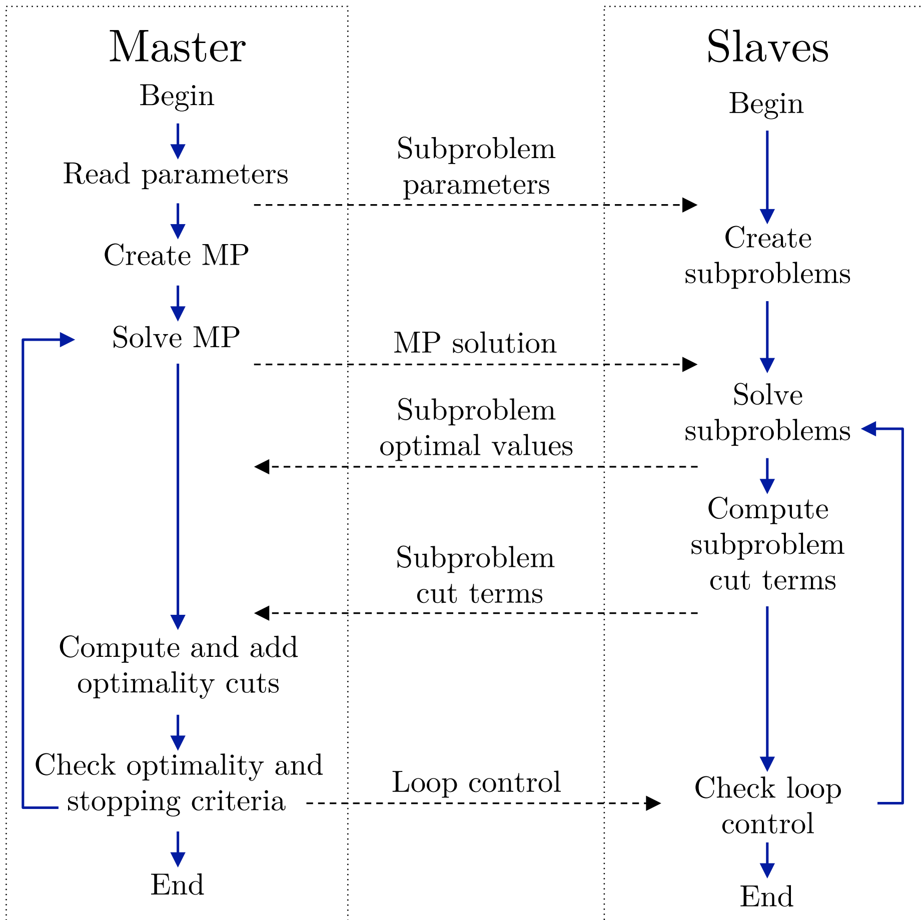


Figure 5.4 Simplified representation of the parallel Benders decomposition algorithm, implemented with MPI

5.5 Computational experiments

In this section we select the combination of acceleration techniques with the best performance on a set of test instances and we evaluate the impact of the parallelization on the computational times of the decomposition algorithm. In these experiments, a treatment corresponds to a combination of acceleration techniques or to a specific configuration of one of them.

5.5.1 Selection of acceleration techniques

For this section, we used a testbed of 24 instances adapted from a real hydropower system in Canada, with the attributes in Table 5.3. Each instance corresponds to a SGMSP with 30 inflow scenarios, 4 powerhouses, 15 time periods and 6 to 8 maintenance tasks. For each powerhouse and number of generators, the hydropower function was approximated with 30 hyperplanes in (5.17).

Table 5.3 Basic attributes of the hydropower system. Powerhouses are ordered from upstream to downstream.

System type	Number of generators	Installed capacity (MW)
Reservoir	5	205
Run of the river	5	210
Reservoir	12	402
Run of the river	17	1587
Total	39	2404

The decomposition algorithm was executed in parallel on a 200-core computer cluster, with one thread dedicated to each subproblem and with up to 10 threads for solving the master problem on an Intel® Xeon® computer at 2.7 GHz.

Since the computational times can differ significantly between instances, we defined as a performance metric the normalized time $\bar{t}_{jb}$ per instance

$$\bar{t}_{jb} = \frac{t_{jb} - \mu_j}{\sigma_j}, \tag{5.54}$$

where t_{jb} is the computational time of the instance $j \in \mathcal{J}$ on treatment $b \in \mathcal{B}$, and μ_j, σ_j are respectively, the mean and standard deviation of the computational times of instance $j \in \mathcal{J}$ in all treatments.

Best combination of acceleration methods

Since the first 7 techniques of Section 5.4.1 can be combined in $2^7 = 128$ different ways, we are interested in identifying which combination has the lowest average computational time. For this purpose, we ran two experiments in sequence. In the first experiment, we applied each of the 7 techniques individually: Valid Inequalities (VI), Warm Start (WS), Multiphase Relaxation (MR), Special Ordered Sets (SOS), Combinatorial Cuts (CC), Pre-solve (PS) and Integer Rounding Cuts (IRC). In this experiment, MR is the relaxation sequence 4, and VI is the combination of valid inequalities (5.43) and (5.44), which reached the smallest computational time in preliminary tests (see Appendix B in Section 5.8.2).

As shown in Fig. 5.5 and Table 5.4, WS achieved the lowest computational times, followed by PS and SOS. Through one-sided t-tests against the basic method, we confirmed that the effect of these three acceleration techniques was highly significant on the computational time (p-

value < 0.001 in Table 5.4). From these results, we fixed, as part of the basic configuration,

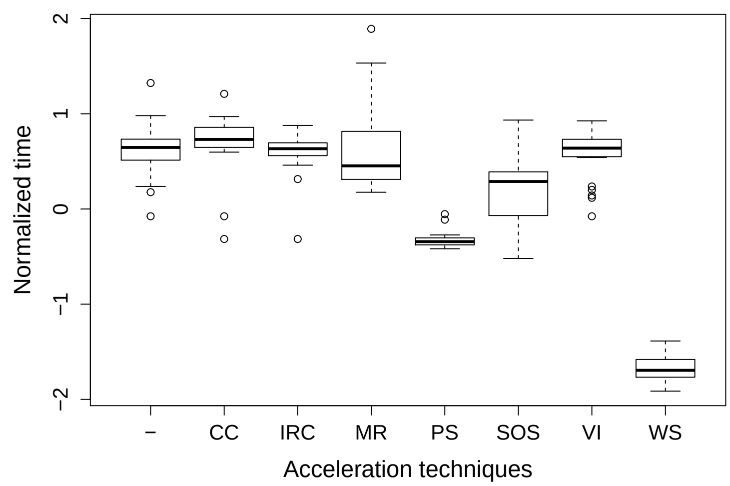


Figure 5.5 Boxplots of normalized computational times of 7 acceleration techniques and the basic method

Table 5.4 Summary statistics of the acceleration methods applied independently. The column *Diff.* shows the difference between the mean time of each technique and the mean time of the basic method (first row).

Treatment	Mean	Std.Dev.	Diff.	p-value
-	0.62	0.27	0.00	-
CC	0.69	0.31	0.07	0.81
IRC	0.59	0.23	−0.03	0.37
MR	0.62	0.43	0.00	0.53
PS	−0.32	0.09	−0.94	6.7e-16
SOS	0.22	0.37	−0.40	5.5e-05
VI	0.56	0.25	−0.06	0.24
WS	−1.68	0.14	−2.30	2.2e-16

the techniques with the lowest computational time (PS, SOS and WS). For selecting the final configuration, we ran a full factorial experiment with the remaining 4 techniques: CC, IRC, MR and VI, which corresponds to $2^4 = 16$ treatments. As shown in Table 5.5, an analysis of variance (ANOVA) applied to the results of this experiment indicates that CC and IRC had a significant effect (p-value < 0.05) on decreasing the computational time ($\beta < 0$), while MR had the opposite effect and VI was not statistically significant. Therefore, in a second ANOVA, we considered only the factors CC and IRC and their interaction term CC·IRC

(Table 5.6). This ANOVA showed that the effects of CC and IRC were statistically significant (p-value < 0.01) on reducing the computational time ($\beta < 0$). Notice that the main effects of CC and IRC (with estimates -0.996 and -0.339, respectively) dominate interaction term CC·IRC (with estimate 0.242), which was not statistically significant (p-value 0.169).

Table 5.5 Summary of linear regression model with techniques VI, MP, CC and IRC as main factors, with normalized computational time as response variable.

	β estimate	p-value
(Intercept)	0.288	0.003
VI	0.089	0.295
MP	0.428	7.6e-07
CC	-0.875	$<$ 2e-16
IRC	-0.218	0.011

Table 5.6 Summary of linear regression model with factors CC and IRC and interaction term. Normalized computational time as response variable

	β estimate	p-value
(Intercept)	0.607	1.9e-11
CC	-0.996	1.2e-14
IRC	-0.339	0.006
CC·IRC	0.242	0.169

From these results, and the previously selected acceleration techniques (Table 5.4), we determined that the recommended combination of the acceleration techniques for the considered problem is: PS, SOS, WS, CC and IRC. In additional tests, this configuration achieved speedups of up to 4 times, with respect to the basic Benders decomposition approach.

5.5.2 Effect of parallelization

For the operation of hydropower systems, as many as 3000 scenarios can be generated to represent the uncertainty of the water inflows (Séguin et al., 2017). In the SGMSP, a large number of scenarios should also be considered to achieve high-quality solutions. Nevertheless, due to the increase in the problem size, a compromise on the number of scenarios must be accepted in practice, depending on the available computational resources and the time limit for obtaining solutions. As a practical example, we consider a case with data adapted from a real 4-powerhouse system, with 8 maintenance tasks to be completed in a planning horizon of 15 days. As in the previous section, for each powerhouse and number of generators, the power production function was approximated with 30 hyperplanes.

We used the same computer cluster and computing server as in Section 5.5.1 for solving the subproblems, and the master problem. To avoid overlapping of subproblems on the 200 available threads, we considered a maximum of 200 scenarios, with 1 subproblem for each thread. With a time limit of 1000 seconds, the decomposition method was benchmarked against the straightforward MILP solution approach, i.e., solving model (5.4)-(5.21) with the MILP solver Xpress-MP. To observe the effect of the number of scenarios on the computational times, we kept constant all the problem parameters, except the size and composition of the set of inflow scenarios. From an initial set of 3028 scenarios, we randomly sampled 12 sets of 200 scenarios each, and we ran tests with 1, 50, 100, 150 and 200 scenarios of each set.

The results indicate that above some point between 50 to 100 scenarios, the parallel Benders decomposition with acceleration techniques outperformed the computational time of the solution with a MILP solver (Fig. 5.6). Furthermore, in instances with 150 and 200 scenarios, the MILP solver reached the 1000-second time limit, with average optimality gaps of 4.6 % and 6.3 %, respectively, while the Benders decomposition approach reached optimal solutions in less than 800 seconds (Fig. 5.6). The results also confirm that, in contrast with the MILP-based solution, the parallel Benders decomposition method is highly scalable. For example, between 50 to 100 scenarios the computational time of the MILP approach increased by 231.7 %, while the computational time via parallel Benders decomposition increased only by 11.5 % (Table 5.7).

The need for considering a sufficiently large number of scenarios is apparent in Table 5.8, where the objective values of the optimization model tend to converge as the number of scenarios increases. For example, in Table 5.8, the variability of the objective values in instances with 150 scenarios (St. dev. 121.6) is less than a half of the variability corresponding to 50 scenarios (St. dev. 267.8). Naturally, this reduction of the variability leads to a better estimate of the actual objective function value.

5.6 Conclusions and future work

We developed a two-stage stochastic program for the hydropower generator maintenance scheduling problem, with binary scheduling decisions in the first stage, and hydropower operation decisions in the second stage. This formulation incorporates relevant aspects of hydropower systems, such as the nonlinearity of hydroelectric production and the uncertainty

Table 5.7 Statistics on the computational times with parallel Benders decomposition and MILP-based solution, with different numbers of inflow scenarios

Number of scenarios	Benders Time		MILP Time	
	Mean	St. dev.	Mean	St. dev.
1	338.9	14.0	0.9	0.3
50	421.2	15.4	213.0	14.0
100	469.8	11.3	706.5	86.0
150	616.5	24.0	-	-
200	780.7	11.8	-	-

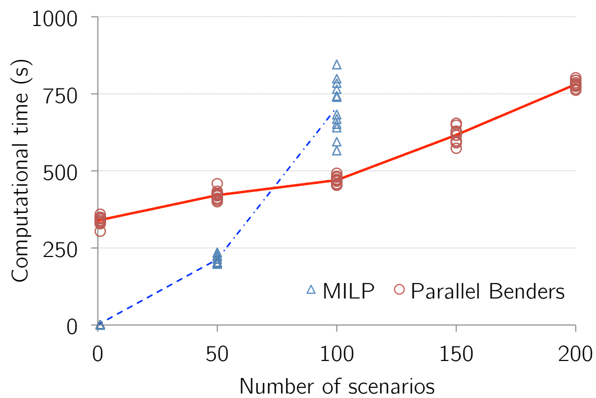


Figure 5.6 Computational time of solving the SGMSP with a MILP solver and with Benders decomposition

of the water inflows. Furthermore, we derived necessary conditions on the problem parameters for a feasible solution.

To solve instances with a large number of inflow scenarios, we implemented a Benders decomposition method, and we tested 7 techniques for accelerating its execution: valid inequalities (VI), warm start (WS), multi-phase relaxation (MR), special ordered sets (SOS), combinatorial cuts (CC), presolve (PS) and integer rounding cuts (IRC). Using statistical methods such as experimental design and analysis of variance, we found that the decomposition algorithm with the combination of PS, SOS, WS, CC and IRC reached the lowest computational time, among the explored combinations. This combination of acceleration techniques achieved speedups of up to 4 times with respect to the basic Benders decomposition approach. Using the MPI protocol, we parallelized the decomposition algorithm for its execution on a computing server and a 200-core computer cluster. In tests with up to 200 scenarios, we confirmed the high scalability of the parallelization on the number of scenarios.

Table 5.8 Mean, standard deviation and 95 % confidence interval of the objective function values, with 12 replicates for each number of scenarios.

Number of scenarios	Objective function value		
	Mean	St. dev.	95% CI
1	13702.6	1799.8	[12559.1, 14846.2]
50	13418.9	267.8	[13248.7, 13589.0]
100	13511.1	211.5	[13376.8, 13645.5]
150	13496.9	121.6	[13419.6, 13574.1]
200	13516.6	100.9	[13452.5, 13580.7]

Future work should address further refinements to the decomposition approach for this problem, such as cut stabilization methods (Fischetti et al., 2016a), and branch-and-Benders-cut (Fortz and Poss, 2009). Alternative decomposition approaches, in combination with constraint programming, are also potential directions for future research.

5.7 Acknowledgments

This research was funded by Rio Tinto, NSERC and MITACS.

5.8 Appendices

5.8.1 Appendix A: Model supplement

A.1 Set reduction

In Rodríguez et al. (2017), the set of numbers of generators is defined as

$$\mathcal{K}(i,t) = \left\{ k \in \mathbb{Z} : \underline{K}_{it} \leq k \leq \bar{K}_{it} \right\}, \forall (i,t) \in \mathcal{I} \times \mathcal{T} \tag{5.55}$$

where

$$\underline{K}_{it} = \max\{\bar{G}_{it} - O_{it}, \bar{G}_{it} - \bar{R}_{it}\}, \tag{5.56}$$

$$\bar{K}_{it} = \bar{G}_{it} - \underline{R}_{it}. \tag{5.57}$$

In (5.56)-(5.57), $\bar{G}_{it}$ denotes the maximum number of available generators at $(i,t) \in \mathcal{I} \times \mathcal{T}$, O_{it} is the maximum number of maintenance outages, and $\bar{R}_{it}$, $\underline{R}_{it}$ denote, respectively, the maximum and minimum number of activities simultaneously in execution at (i,t), according to their time windows, i.e.,

$$\underline{R}_{it} = \big|\{\, (m,t) \in \mathcal{M}(i) \times \mathcal{T}(m) \;:\; L_m \leq t \leq E_m + D_m - 1 \,\}\big|, \tag{5.58}$$

$$\bar{R}_{it} = \big|\{\, (m,t) \in \mathcal{M}(i) \times \mathcal{T}(m) \;:\; E_m \leq t \leq L_m + D_m - 1 \,\}\big|, \tag{5.59}$$

where for each activity $m \in \mathcal{M}$, we denote by D_m, E_m and L_m its duration, earliest starting time and latest starting time, respectively.

A.2 Conditions for feasible subproblems

From the viewpoint of computational efficiency, *complete recourse* and *relatively complete recourse* are desirable properties of stochastic programming problems (Birge and Louveaux, 2011). In problems with these properties, the Benders decomposition method will only generate feasible solutions at each iteration. A stochastic program is said to have *complete recourse* if the second-stage problem (i.e., the subproblem) is always feasible. If the stochastic program has *relatively complete recourse*, the second-stage problem is feasible for any feasible first-stage solution and scenario realization. Following these definitions, we notice that the subproblem (5.25)-(5.35) has *partially complete recourse* (i.e., is feasible for any inflow scenario and master problem feasible solution), if the following conditions are met:

1. The system (5.26), (5.31)-(5.33) is feasible for any inflow realization $\xi_{it\omega}$, where $(i,t,\omega) \in \mathcal{I} \times \mathcal{T} \times \Omega$.
2. In all time periods, the electricity load A_t is not greater than the upper bound of the electricity purchase, i.e., $0 \leq A_t \leq \bar{W}_t^-$, $\forall\, t \in \mathcal{T}$.

Without loss of generality, we assume that the instances of the SGMSP satisfy conditions 1 and 2. Notice that these conditions can be guaranteed with proper values of the variable bounds (5.32)-(5.35). If either of these conditions are not met, it would be necessary to include feasibility cuts at some iterations of the Benders algorithm. Alternatively, the partial complete recourse property can be reestablished with the introduction of artificial variables in (5.26), (5.29), and with a penalization of these variables in the objective function (5.25).

A.3 Valid inequalities

1. The first family of valid inequalities comes from the observation in Rodríguez et al. (2017) that in a powerhouse i, if at least one maintenance task $m \in \mathcal{M}(i)$ is in execution at time t, then the binary variable corresponding to $\bar{G}_{it}$ active generators must be equal

to zero, i.e., $z_{itk} = 0$, for $k = \bar{G}_{it}$. Thus,

$$\sum_{\substack{m \in \mathcal{M}(i) \\ t' \in \mathcal{T}(m) \cap [\, t - D_m + 1,\, t\,]}} y_{mt'} + z_{itk} \leq 1, \quad \text{for } k = \bar{G}_{it}, \forall \; (i,m,t) \in \mathcal{I} \times \mathcal{M}(i) \times \mathcal{T}, \tag{5.60}$$

are valid inequalities. Naturally, such inequalities are unnecessary when $\bar{K}_{it} < \bar{G}_{it}$ (5.55) or when the set $t' \in \mathcal{T}(m) \cap [\, t - D_m + 1,\, t\,]$ is empty.

2. The second family of valid inequalities comes from the fact that for any (i,t), when the number of maintenance outages is zero, i.e., $r_{it} = 0$, then all $\bar{G}_{it}$ generators are active ($z_{itk} = 1$, for $k = \bar{G}_{it}$) (Rodríguez et al., 2017). By (5.9), it follows that $z_{itk} = 0$ for $k < \bar{G}_{it}$, which is equivalent to

$$\sum_{k \in \mathcal{K}(i,t) \setminus \{\bar{G}_{it}\}} z_{itk} \; \leq r_{it}, \; \forall \; (i,t) \in \mathcal{I} \times \mathcal{T}. \tag{5.61}$$

Such inequalities are also unnecessary when $\bar{K}_{it} < \bar{G}_{it}$.

3. From (5.56) we notice that

$$\bar{G}_{it} \leq \underline{K}_{it} + \bar{R}_{it}, \quad (i,t) \in \mathcal{I} \times \mathcal{T}. \tag{5.62}$$

Then, applying (5.62) on the left-hand side of (5.8) gives

$$r_{it} + \sum_{k \in \mathcal{K}(i,t)} k z_{itk} \; \leq \underline{K}_{it} + \bar{R}_{it}, \; \forall \; (i,t) \in \mathcal{I} \times \mathcal{T},$$

which by (5.9) and (5.55) leads to

$$r_{it} + \sum_{k \in \mathcal{K}(i,t) \setminus \{\underline{K}_{it}\}} (k - \underline{K}_{it}) z_{itk} \; \leq \bar{R}_{it}, \; \forall \; (i,t) \in \mathcal{I}, \times \mathcal{T}. \tag{5.63}$$

A.4 Upper bound of subproblem objective value

If the assumption $A_t \leq \bar{P}_{ik} \leq W_t$ does not hold, (5.45) can be replaced with

$$
\begin{aligned}
z^{SP} \leq \sum_{t \in \mathcal{T}} B_t^+ \Bigg(& \sum_{\substack{i \in \mathcal{I}, \\ k \in \{\mathcal{K}(i,t): \\ \bar{P}_{ik} > A_t \,;\, W_t^+ > \bar{P}_{ik} - A_t\}}} (\bar{P}_{ik} - A_t) z_{itk} \\
& + \sum_{\substack{i \in \mathcal{I}, \\ k \in \{\mathcal{K}(i,t): \\ \bar{P}_{ik} > A_t \,;\, W_t^+ < \bar{P}_{ik} - A_t\}}} W_t^+ z_{itk} \Bigg) \\
& - \sum_{t \in \mathcal{T}} B_t^- \Bigg(\sum_{\substack{i \in \mathcal{I}, \\ k \in \{\mathcal{K}(i,t): \\ \bar{P}_{ik} < A_t \}}} (A_t - \bar{P}_{ik}) z_{itk} \Bigg),
\end{aligned}
\tag{5.64}
$$

where the first term is the maximum sold electricity when the electricity surplus is less than the bound of the electricity sale. The second term is the maximum sold electricity when the bound on the electricity sale is less than the electricity surplus, and the third term is the cost of the electricity purchase when the load exceeds the generation capacity.

5.8.2 Appendix B: Selecting multiple-phase relaxation sequence and valid inequalities

B.1 Valid Inequalities

On the set of 24 instances, we ran a factorial experiment with the $2^3 = 8$ combinations of the three families of valid inequalities of Section 5.4.1. To select the best combination of these inequalities, we sequentially applied analysis of variance (ANOVA) with normalized computational time as the response variable. From the results of the first ANOVA, with each family of valid inequalities defined as a categorical factor (Table 5.9), we dropped the valid inequality family 1 (factor VI1) for increasing the computational times ($\beta = 0.188$) at a significance level of 0.1 (p-value $= 0.055$). With the same experimental data, an ANOVA with the factors VI2 and VI3 and the interaction term VI2·VI3 (see Table 5.10) shows that the combination of the valid inequalities 2 and 3 (i.e., the interaction term VI2·V3) has the lowest average computational time ($\beta = -0.363$), at a significance level of 0.1 (p-value $=$ 0.064).

Table 5.9 Summary of ANOVA with valid inequalities 1, 2 and 3 as main factors, and normalized computational time as response variable.

	β estimate	p-value
(Intercept)	0.078	0.427
VI1	0.188	0.055
VI2	-0.095	0.333
VI3	-0.249	0.011

Table 5.10 Summary of ANOVA with valid inequalities 1 and 2 and interaction term, and normalized computational time as response variable.

	β estimate	p-value
(Intercept)	0.081	0.408
VI2	0.087	0.531
VI3	-0.067	0.627
VI2·VI3	-0.363	0.064

B.2 Multiple-phase relaxation

We defined the relaxation sequences of Table 5.2 as treatments. In these sequences, each phase is completed when either a specified maximum number of cuts or a maximum optimality gap is reached (Table 5.11). According to the results, the sequence without relaxation (i.e.,

Table 5.11 Parameters of stages in multi-phase relaxation.

Relax. level	Binary var.	Max. cuts	Max. gap
0	y, z	1000	1.0e-5
1	y	4	0.005
2	z	4	0.005
3	-	20	0.010

relaxation sequence 0), exhibited the largest variability and the highest computational time (Fig. 5.7). An analysis of variance on the 24 instances indicated that the multi-phase relaxation had a significant effect on the computational times (p-value $= 0.00924$). Although the computational times of the relaxation sequences 3, 4 and 5 were similar, the relaxation sequence 4 showed the most significant effect (p-value $= 0.007$) in a one-tailed t-test against the method without relaxation (see Table 5.12). Therefore, the best configuration applies the relaxation sequence $(y, z) \rightarrow (z) \rightarrow (y)$, before solving the master problem without relaxation.

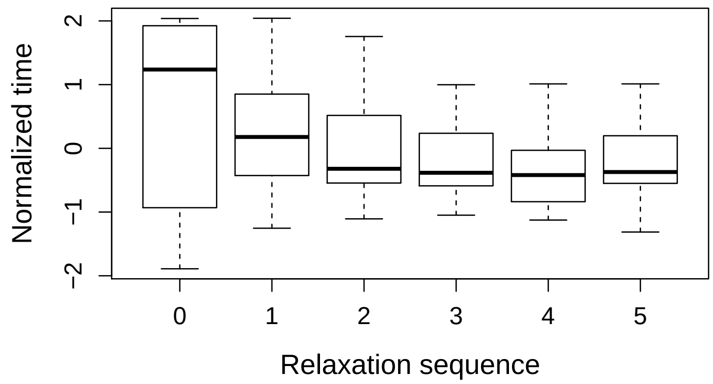


Figure 5.7 Boxplot of the computational times of the multi-phase relaxation sequences on 24 instances.

Table 5.12 Summary statistics of normalized computational times of multi-phase relaxations. The column *Diff.* shows the difference between the mean time of each sequence and the mean of sequence 0.

Relax. Seq.	Mean	St. Dev.	Diff.	p-value
0	0.54	1.51	0.00	-
1	0.21	0.91	−0.33	0.181
2	−0.04	0.71	−0.58	0.048
3	−0.19	0.57	−0.73	0.017
4	−0.34	0.63	−0.88	0.007
5	−0.18	0.60	−0.72	0.019

5.8.3 Appendix C: Nomenclature

Primary sets	
$\mathcal{I}$	Powerhouses
$\mathcal{M}$	Maintenance tasks
$\mathcal{T}$	Planning time periods, $t \in \mathcal{T} = \{1 \dots T\}$
Ω	Scenarios

Parameters	
$\xi_{it\omega}$	Lateral inflows to powerhouse i in period t and scenario ω, [m^3/s].
A_t	Electricity load at time period t.
B_t^+	Electricity sale price in time period t, [\$/MWh].
B_t^-	Electricity purchase price in time period t, [\$/MWh].
C_{mt}	Total cost of maintenance task m started at time period t, [\$].
D_m	Duration of maintenance task m [day].
E_m	Earliest start time period of maintenance task m.
F	Factor for conversion from flow per second in m^3 to flow per day in hm^3 [0.0864·s·hm^3 ·/(day'·m^3)].
$\bar{G}_{it}$	Maximum number of available turbines in powerhouse i at time period t, [turbines].
$\underline{G}_i$	Minimum number of available turbines in powerhouse i [turbines].
L_m	Latest start time period of maintenance task m.
O_{it}	Maximum number of turbine outages in powerhouse i at time period t, [turbines].
$\bar{P}_i$	Generation capacity in powerhouse i, [MWh/day].
$\bar{P}_{ik}$	Generation capacity in powerhouse i when k turbines are active, [MWh/day].
$Q(\bar{y})$	Expected operating cost of solution $\bar{y}$ [\$].
$Q_\omega(\bar{y})$	Expected operating cost of solution $\bar{y}$ in scenario ω [\$].
$\bar{R}_{it}$	Number of maintenance activities that *can* be in execution at powerhouse i in time period t.
$\underline{R}_{it}$	Number of maintenance activities that *must* be in execution at powerhouse i in time period t.
S_{0i}	Initial volume in reservoir of powerhouse i, [hm^3].
$\underline{S}_i, \bar{S}_i$	Limits on stored water in reservoir of powerhouse i at period t [hm^3].
$\bar{U}_{it}$	Maximum discharge rate in powerhouse i, [m^3/s].
$\bar{V}_{it}$	Maximum water spill in powerhouse i, [m^3/s].
$\bar{W}_t^+$	Maximum electricity sale at time t [MWh/day].
$\bar{W}_t^-$	Maximum electricity purchase at time t [MWh/day].

Derived sets	
$\mathcal{T}(m)$	Time periods when maintenance task m can be initiated in order to be completed within $\mathcal{T}$.
$\mathcal{M}(i)$	Maintenance tasks m that should be executed in powerhouse i.
$\mathcal{M}(i,t)$	Maintenance tasks m that can be in execution in powerhouse i at time period t.
$\mathcal{U}(i)$	Powerhouses upstream of powerhouse i ($\mathcal{U}(i) \subset \mathcal{I}$).
$\mathcal{K}(i,t)$	Numbers of generators that can be active at time period t and powerhouse i.
$\mathcal{H}(i,k)$	Hyperplanes for approximating the maximum power of powerhouse i when k turbines are active.
$\mathcal{A}$	set of indices (m,t) of variables y_{mt} with value 1 in solution $\bar{y}$, i.e, $\mathcal{A} = \{(m,t) \in \mathcal{M} \times \mathcal{T} \mid \bar{y}_{mt} = 1\}$.
Parameters with indexes in derived sets	
β_h^u	Coefficient of u_{it} in hyperplane $h \in \mathcal{H}(i,k)$ for bounding the power output of powerhouse i when k generators are active [MWh· s/(m^3·day)].
β_h^s	Coefficient of s_{it} in hyperplane $h \in \mathcal{H}(i,k)$ for bounding the power output of powerhouse i when k generators are active [MWh/(hm^3·day)].
β_h^0	Independent term of hyperplane $h \in \mathcal{H}(i,k)$ for bounding the power output of powerhouse i when k generators are active [MWh/day].
Decision variables	
$p_{it\omega}$	Generation of powerhouse i during time period t in scenario ω [MWh/day].
$p_{itk\omega}$	Generation of powerhouse i during time period t in scenario ω when k generators are active [MWh/day].
$q_{t\omega}^+$	Sale of electricity at period t in scenario ω [MWh].
$q_{t\omega}^-$	Purchase of electricity at period t in scenario ω [MWh].
r_{it}	Number of maintenance activities in execution at powerhouse i and time period t.
$s_{it\omega}$	Content of reservoir in powerhouse i at the end of period t in scenario ω [hm^3].
$u_{it\omega}$	Water discharge of turbines in powerhouse i at time period t in scenario ω [m^3/s].
$v_{it\omega}$	Water spill of reservoir in powerhouse i at time period t in scenario ω [m^3/s].
y_{mt}	Binary variable with value 1 if maintenance task m initiates at time period t, 0 otherwise.
z_{itk}	Binary variable with value 1 if k hydro-turbines are active in powerhouse i at time t, 0 otherwise.
z^{SP}	Approximated expected profit of the hydroelectric production [$].
z_ω^{SP}	Profit of the hydroelectric production in scenario ω [$].

Dual variables	
$\pi^{p}_{it\omega}$	of mass balance constraint (5.26) in solution p.
$\gamma^{p}_{itkh\omega}$	of power function (5.27) in solution p.
$\lambda^{p}_{itk\omega}$	of power bound constraint (5.28) in solution p.
$\psi^{p}_{t\omega}$	of power balance constraint (5.29) in solution p.
$\theta^{p}_{it\omega}$	of sum of power constraint (5.30) in solution p.
$\alpha^{pu}_{it\omega}$	of water discharge bound (5.32) in solution p
$\alpha^{ps}_{it\omega}$	of stored water bounds (5.33) in solution p
$\alpha^{p+}_{t\omega}$	of electricity sale bounds (5.34) in solution p
$\alpha^{p-}_{t\omega}$	of electricity purchase bounds (5.35) in solution p

CHAPTER 6 EXTENSIONS

In this chapter, we propose an alternative decomposition strategy for the SGMSP, and we enhance our mixed-integer programming formulation for this problem with several types of maintenance constraints and with additional decision variables.

6.1 Model extensions

Among the multiple possibilities for enhancing the mixed-integer formulation for the SGMSP, we discuss three types of extensions:

— Considering additional maintenance scheduling constraints, such as precedence of maintenance activities and available maintenance resources.

— Selecting the duration of maintenance tasks, given a discrete set of duration alternatives.

— Including reserves of energy and capacity to buffer the forecast errors of the electricity load.

6.1.1 Additional maintenance constraints

Our formulation for the SGMSP in Chapter 5, included only basic maintenance constraints, namely, completion of maintenance tasks, maximum number of maintenance outages, maintenance time-windows and mapping of number of active generators. We used index sets for a compact definition of such constraints. In the general case, other maintenance constraints may also be necessary, such as:

— Available maintenance resources
— Mutually exclusive tasks
— Overlapping of maintenance activities
— Precedence of starting times
— Precedence of maintenance execution

Next we show that index sets can also compactly define such constraints.

Available maintenance resources

Because maintenance resources such as workforce and equipment have a finite availability, their use must be controlled at each time period $t \in \mathcal{T}$. Let $\mathcal{N}$ be the set of maintenance resources with index n and time-variant availability Γ_{nt} (in units of resource n). Let A_{mn} be the use of maintenance resource n (also in units of resource n) per time period of execution of maintenance task $m \in \mathcal{M}$, and let w_{nt} denote the additional units of resource n consumed at time t, at a cost Υ_{nt} per unit. Then the following constraints control the consumption of the maintenance resources

$$\sum_{\substack{m \in \mathcal{M} \\ t' \in \mathcal{T}(m) \cap [t - D_m + 1, t]}} A_{mn} y_{mt'} - w_{nt} \leq \Gamma_{nt}, \quad \forall\, (t, n) \in \mathcal{T} \times \mathcal{N}, \tag{6.1}$$

$$w_{nt} \geq 0, \quad \forall\, (t, n) \in \mathcal{T} \times \mathcal{N}, \tag{6.2}$$

where the index set of the summation on the left hand-side of (6.1) includes into the constraint only the decision variables of the maintenance activities that started at time period $t' \leq t$ and that are still in execution at t. The consumption of extra maintenance resources computed by means of (6.1) and (6.2) must be penalized by including into the objective function (5.21) the cost term

$$\sum_{(t,n) \in \mathcal{T} \times \mathcal{N}} \Upsilon_{nt} w_{nt}. \tag{6.3}$$

Mutually exclusive tasks

For technical or administrative reasons, some maintenance tasks may not be allowed to be in execution at the same time. Let $\mathcal{E}$ be the set of pairs of mutually exclusive activities. To prevent $(m_1, m_2) \in \mathcal{E}$ from occurring at the same time, we define the constraints

$$\sum_{t' \in \mathcal{T}(m_1) \cap [t - D_{m_1} + 1, t]} y_{m_1 t'} + \sum_{t' \in \mathcal{T}(m_2) \cap [t - D_{m_2} + 1, t]} y_{m_2 t'} \leq 1,$$
$$\forall\, (m_1, m_2) \in \mathcal{E},\ t \in \mathcal{T}^{\mathcal{S}}(m_1) \cap \mathcal{T}^{\mathcal{S}}(m_2), \tag{6.4}$$

where, for an activity m, D_m is its duration, $\mathcal{T}(m)$ is its feasible interval of starting times, and $\mathcal{T}^{\mathcal{S}}(m)$ is its span, i.e., the time interval when the activity m can be in execution (see Section 4.3.3).

In (6.4), each of the summations on the left-hand side is equal to 1 if the corresponding activity is in execution at time period t. Thus, as the activities $(m_1, m_2) \in \mathcal{E}$ cannot be simultaneously in execution, the sum on the left-hand side of (6.4) must be less than or equal to 1. Naturally, for two mutually exclusive tasks (m_1, m_2), constraints (6.4) are only necessary if the spans of the activities overlap, i.e., if $\mathcal{T}^{\mathcal{S}}(m_1) \cap \mathcal{T}^{\mathcal{S}}(m_2) \neq \emptyset$.

Overlapping of maintenance activities

If a maintenance task m_2 can only start while the maintenance task m_1 is still in execution, we say that the pair of activities (m_1, m_2) must overlap with precedence of m_1 over m_2. The following constraints control this condition

$$y_{m_2 t} \leq \sum_{t' \in \mathcal{T}(m_1) \cap [t - D_{m_1} + 1, t]} y_{m_1 t'}, \quad \forall (m_1, m_2) \in \mathcal{O}, t \in \mathcal{T}(m_2), \tag{6.5}$$

where $\mathcal{O}$ is the set of pairs of overlapping activities. Constraints (6.5) allow maintenance task m_2 to be started at t only when the summation on the right-hand side is equal to 1, indicating that m_1 is in execution. Notice that overlapping of (m_1, m_2) is only feasible if $\mathcal{T}^{\mathcal{S}}(m_1) \cap \mathcal{T}(m_2) \neq \emptyset$. Furthermore, for the precedence of m_1 over m_2 in (6.5), we assume that m_1 can start before m_2, i.e., $E_{m_1} \leq E_{m_2}$.

Precedence of starting times

Let $\mathcal{P}_1$ be the set of pairs of activities $(m_1, m_2) \in \mathcal{M} \times \mathcal{M}$, such that the starting time of m_1 precedes the starting time of m_2, with $E_{m_1} \leq E_{m_2}$. The precedences of $\mathcal{P}_1$ can be enforced through the constraints

$$y_{m_2 t} \leq \sum_{t' \in \mathcal{T}(m_1) \cap [E_{m_1}, t-1]} y_{m_1 t'}, \quad \forall (m_1, m_2) \in \mathcal{P}_1, t \in \mathcal{T}(m_1) \cap \mathcal{T}(m_2). \tag{6.6}$$

Notice that m_2 can start at time period t only when the sum on the right hand-side of (6.6) is equal to 1, which indicates that the activity m_1 has started before t. Due to the assumption on the earliest starting times of the activities (i.e., $E_{m_1} \leq E_{m_2}$), $y_{m_2 t}$ is not defined for $t \leq E_{m_1}$, and for $t \in \mathcal{T}(m_2) \setminus \mathcal{T}(m_1)$, $y_{m_2 t}$ satisfies the precedence. Thus, constraints (6.6) only need to be defined for the time interval $\mathcal{T}(m_1) \cap \mathcal{T}(m_2)$.

Precedence of maintenance execution

Let $\mathcal{P}_2$ denote the set of pairs of activities $(m_1, m_2) \in \mathcal{M} \times \mathcal{M}$, such that activity m_1 must be completed before the starting time of m_2, with $E_{m_1} \leq E_{m_2}$. The precedences $\mathcal{P}_2$ are implied by the constraints

$$y_{m_2 t} \leq \sum_{t' \in \mathcal{T}(m_1) \cap [E_{m_1}, t - D_{m_1}]} y_{m_1 t'}, \quad \forall (m_1, m_2) \in \mathcal{P}_2, t \in \mathcal{T}^S(m_1) \cap \mathcal{T}(m_2), \tag{6.7}$$

where at each time period t the summation term on the right hand-side is equal to 1 if the activity m_1 has started in the interval $\mathcal{T}(m_1) \cap [\, E_{m_1},\, t - D_{m_1}\,]$, and thus has completed its execution before t. Due to the assumption $E_{m_1} \leq E_{m_2}$, constraints (6.7) are only defined for the interval $\mathcal{T}^S(m_1) \cap \mathcal{T}(m_2)$.

6.1.2 Selecting the duration of maintenance activities

In several planning and scheduling problems, durations of activities are not fixed parameters but choices among sets of discrete alternatives. Trade-offs of such alternatives arise, as expedited activities, usually with high execution costs, can reduce downtime and speed-up the achievement of the objectives. Let $\mathcal{D}(m)$ be the index set of possible durations of an activity $m \in \mathcal{M}$. To consider such alternative durations, we define the binary variables

$$y_{mdt} \in \{0, 1\},\ \forall\ (m, d, t) \in \mathcal{M} \times \mathcal{D}(m) \times \mathcal{T}(m), \tag{6.8}$$

where y_{mdt} is equal to 1 if maintenance task m with duration index d starts at time period t, and 0 otherwise. Given the duration parameter D_{md} of each activity m with duration index d, we define the constraints

$$\sum_{\substack{d \in \mathcal{D}(m) \\ t \in \mathcal{T}(m)}} y_{mdt} = 1, \quad \forall\ m \in \mathcal{M}, \tag{6.9}$$

$$\sum_{\substack{m \in \mathcal{M}(i) \\ d \in \mathcal{D}(m) \\ t' \in \mathcal{T}(m) \cap [\, t - D_{md} + 1,\, t\,]}} y_{mdt'} = r_{it}, \quad \forall\ (i, t) \in \mathcal{I} \times \mathcal{T}, \tag{6.10}$$

and the objective function

$$\underset{\substack{q^+, q^-, u, v, s, \\ r, p, y, z}}{\text{maximize}} \ \sum_{\substack{t \in \mathcal{T} \\ \omega \in \Omega}} \varphi_\omega \Big(B_t^+ q_{t\omega}^+ - B_t^- q_{t\omega}^- \Big) - \sum_{\substack{m \in \mathcal{M} \\ d \in \mathcal{D}(m) \\ t \in \mathcal{T}(m)}} C_{mdt} y_{mdt}, \tag{6.11}$$

where (6.9) ensures the completion of the maintenance activities, with exactly one duration and time period chosen for each activity; (6.10) computes the number of activities simultaneously in execution at each time period t and powerhouse i, i.e., the number of activities $m \in \mathcal{M}(i)$, with duration D_{md}, that started on the interval $\mathcal{T}(m) \cap [\, t - D_{md} + 1,\, t\,]$. The objective function (6.11) maximizes the difference between the economic gains of the hydropower operation and the sum of the maintenance costs, with parameters C_{mdt} for each activity and duration index $d \in \mathcal{D}(m)$. Therefore, the SGMSP with discrete choice of activities' duration is

$$\text{maximize (6.11) subject to (5.5), (5.8)} - \text{(5.20), (6.8)} - \text{(6.10).} \quad (6.12)$$

6.1.3 Load uncertainty and generation reserves

In Chapter 5 we considered the uncertainty of the water inflows through a two-stage stochastic programming formulation with a scenario fan of forecasted inflows. Although this approach can be extended to consider the uncertainty of the electricity load (due to forecast errors), the resulting maintenance schedules can work well on the average but insufficient electricity generation can occur in several scenarios. We discuss chance-constrained and max-min formulations for ensuring generation reserves in maintenance scheduling, to buffer the forecast errors of the electricity load.

Chance-constrained approach

Chance-constrained optimization is an approach for problems in which some constraint violations can be accepted with specified probabilities (Henrion, 2004; Birge and Louveaux, 2011). We apply this approach for considering the uncertainty of the electricity demand.

When the availability and reliability of purchased electricity cannot be guaranteed, electricity producers must ensure sufficient internal capacity and energy reserves to supply its electricity load. Let $\hat{p}_{it\omega}$ denote the electricity generation at powerhouse $i \in \mathcal{I}$, time period $t \in \mathcal{T}$, and inflow scenario $\omega \in \Omega$, for supplying a random realization δ_t^A of the electricity load. Assuming that at each time period the load δ_t^A is normally distributed with an unbiased forecasted value A_t and a forecast error σ_t^A, the constraint

$$\sum_{i \in \mathcal{I}} \hat{p}_{it\omega} \geq \delta_t^A, \quad \forall\ (t, \omega) \in \mathcal{T} \times \Omega, \quad (6.13)$$

can only be satisfied with some probability θ due to the forecast error of the load and the finite energy and generation capacity. That is,

$$\Pr\Big(\sum_{i \in \mathcal{I}} \hat{p}_{it\omega} \geq \delta_t^A\Big) \geq \theta, \quad \forall\ (t, \omega) \in \mathcal{T} \times \Omega. \quad (6.14)$$

In order to linearize (6.14) with individual chance constraints (Henrion, 2004), we apply a standard normalization (Montgomery and Runger, 2010) on their constraint terms inside the probability function. Thus, on each term of the left-hand side, we normalize by subtracting the forecasted electricity load A_t and dividing by the standard error σ_t^A of the forecast, i.e.,

$$\Pr\left(\frac{\sum_{i\in\mathcal{I}}\hat{p}_{it\omega} - A_t}{\sigma_t^A} \geq \frac{\delta_t^A - A_t}{\sigma_t^A}\right) \geq \theta, \quad \forall\ (t,\omega) \in \mathcal{T} \times \Omega. \tag{6.15}$$

Then, by substituting in (6.15) the term ${(\delta_t^A - A_t)}/{\sigma_t^A}$ with a standard normal variable Z, the equivalent constraint

$$\Pr\left(\frac{\sum_{i\in\mathcal{I}}\hat{p}_{it\omega} - A_t}{\sigma_t^A} \geq Z\right) \geq \theta, \quad \forall\ (t,\omega) \in \mathcal{T} \times \Omega,$$

is implied by the constraints

$$\frac{\sum_{i\in\mathcal{I}}\hat{p}_{it\omega} - A_t}{\sigma_t^A} \geq Z_\theta, \quad \forall\ (t,\omega) \in \mathcal{T} \times \Omega,$$

where Z_θ is the standard normal value corresponding to the cumulative probability θ. Therefore, the probabilistic constraints (6.14) are equivalent to

$$\sum_{i\in\mathcal{I}}\hat{p}_{it\omega} \geq A_t + \sigma_t^A Z_\theta, \quad \forall\ (t,\omega) \in \mathcal{T} \times \Omega. \tag{6.16}$$

Notice that due to (6.16) the variables $\hat{p}_{it\omega}$ of the chance-constrained formulation can differ from the original electricity production variables $p_{it\omega}$, which are computed based on the expected electricity load, purchases and sales, as defined in the power balance constraints (5.20). Therefore, we define auxiliary variables and constraints to ensure that the solution values of $\hat{p}_{it\omega}$ for meeting the load requirements in (6.16) are consistent with the stored water levels, the water inflows and the generation capacity:

$$0 \le \hat{v}_{it\omega}, \quad \forall\ (i,t,\omega) \in \mathcal{I} \times \mathcal{T} \times \Omega, \tag{6.17}$$

$$0 \le \hat{u}_{it\omega} \le \bar{U}_{it}, \quad \forall\ (i,t,\omega) \in \mathcal{I} \times \mathcal{T} \times \Omega, \tag{6.18}$$

$$\underline{S}_{it} \le \hat{s}_{it\omega} \le \bar{S}_{it}, \quad \forall\ (i,t,\omega) \in \mathcal{I} \times \mathcal{T} \times \Omega, \tag{6.19}$$

$$\hat{s}_{it\omega} - \hat{s}_{i(t-1)\omega} = \left(\xi_{it} + \sum_{g \in \mathcal{U}(i)} (\hat{u}_{gt\omega} + \hat{v}_{gt\omega}) - \hat{u}_{it\omega} - \hat{v}_{it\omega}\right) F,$$
$$\forall\ (i,t,\omega) \in \mathcal{I} \times \mathcal{T} \times \Omega, \tag{6.20}$$

$$\tag{6.21}$$

$$\hat{p}_{itk\omega} \le \beta_h^0 + \beta_h^u \hat{u}_{it\omega} + \beta_h^s \hat{s}_{it\omega}, \quad \forall\, (i,t,k,h,\omega) \in \mathcal{I} \times \mathcal{T} \times \mathcal{K}(i,t) \times \mathcal{H}(i,k) \times \Omega, \tag{6.22}$$

$$0 \le \hat{p}_{itk\omega} \le z_{itk} \bar{P}_{ik}, \quad \forall\, (i,t,k,\omega) \in \mathcal{I} \times \mathcal{T} \times \mathcal{K}(i,t) \times \Omega, \tag{6.23}$$

$$\sum_{k \in \mathcal{K}(i,t,\omega)} \hat{p}_{itk\omega} = \hat{p}_{it\omega}, \quad \forall\ (i,t) \in \mathcal{I} \times \mathcal{T} \times \Omega, \tag{6.24}$$

where $\hat{v}_{it\omega}$, $\hat{u}_{it\omega}$, $\hat{s}_{it\omega}$ and $\hat{p}_{itk\omega}$ are auxiliary decision variables for the electricity production $\hat{p}_{it\omega}$ under reserve requirements (6.16), and (6.17)-(6.24) are auxiliary operational constraints that correspond one-to-one with the constraints (5.11)-(5.19) of the hydropower operation under expected load conditions (5.20). Therefore the chance-constrained formulation for the SGMSP is

$$\text{maximize (5.21) subject to (5.4)} - \text{(5.20)}, \text{(6.16)} - \text{(6.24)}.$$

Notice that to compute the expected economic benefit of the operation under predicted load conditions, the original variables and constraints of the hydropower operation model are also included in the chance-constrained formulation. To prevent a very large number of variables and constraints in this formulation, only a worst-case water inflow scenario $\hat{\xi}_{it}$ needs to be considered in (6.16)-(6.24). With this simplification, the scenario indices ω can be dropped from the auxiliary variables $\hat{v}_{it\omega}$, $\hat{u}_{it\omega}$, $\hat{s}_{it\omega}$, $\hat{p}_{itk\omega}$, $\hat{p}_{it\omega}$ as well as from the auxiliary constraints (6.16)-(6.24).

Max-min formulation approach

Some previous works on maintenance scheduling in regulated power systems have defined as an optimization criterion levelizing the net reserves, i.e., the difference between the predicted load and the generation capacity (Froger et al., 2016). However, such an approach does not guarantee a robust solution with sufficient reserves in the worst-case scenario and time period. We apply a max-min formulation to maximize the minimum reserve in the SGMSP. Denoting by x the minimum surplus reserve over the planning horizon, we define the objective function

$$\underset{x}{\text{maximize}} \quad x, \tag{6.25}$$

and the constraints

$$\sum_{i\in\mathcal{I}} \hat{p}_{it\omega} \geq A_t + R_t + x, \quad \forall\ (t,\omega) \in \mathcal{T} \times \Omega \tag{6.26}$$

$$x \geq 0, \tag{6.27}$$

where R_t is the specified minimum reserve at time period $t \in \mathcal{T}$. Therefore, the max-min formulation for the SGMSP is

$$\text{maximize } (6.25) \text{ subject to } (5.4) - (5.19), (6.26) - (6.27).$$

Notice that defining $R_t = \sigma_t^A Z_\theta$, the max-min approach ensures that at each time period and scenario the reserve requirements of the chance constrained-approach are met, as in (6.16).

6.2 Alternative Decomposition Strategy (ADS)

As discussed in Section 2.1.3, Benders decomposition is a row generation procedure that splits mathematical programs with complicating variables into a master problem with the complicating variables, and convex subproblems. In problems with diverse types of variables, the master problem and the subproblems can be defined in multiple ways, each of them leading to a different design and execution of the decomposition algorithm.

In Chapter 5 we applied Benders decomposition to the SGMSP with a straightforward partitioning of the problem (i.e., with all the complicating variables in the master problem). In this section, we present an alternative decomposition strategy for this problem, with the master problem involving only a subset of the complicating variables.

To derive such an alternative approach, first we analyze the mathematical structure of the SGMSP using a compact matrix representation. Let x_ω, y, z be vectors with dimensions n_x, n_y and n_z respectively. By x_ω, y and z we denote the decision variables of the hydropower operation, the starting times of maintenance activities and the indicators of the number of active generators, respectively. According to such definitions, the compact representation of the deterministic equivalent of the SGMSP is

$$\max_{y,z,x_\omega} \sum_{\omega \in \Omega} \varphi_\omega c^\intercal x_\omega - f^\intercal y \tag{6.28}$$

subject to

$$A_1 x_\omega \qquad + C_1 z \geq b_{1\omega}, \ \forall \omega \in \Omega \tag{6.29}$$

$$A_2 x_\omega \qquad \geq b_{2\omega}, \ \forall \omega \in \Omega \tag{6.30}$$

$$B_3 y + C_3 z = b_3 \tag{6.31}$$

$$B_4 y \geq b_4 \tag{6.32}$$

$$C_5 z \geq b_5 \tag{6.33}$$

$$x_\omega \geq 0, \quad \forall \omega \in \Omega \tag{6.34}$$

$$y \in \{0,1\}^{n_y} \tag{6.35}$$

$$z \in \{0,1\}^{n_z} \tag{6.36}$$

where the vectors c, b and matrices A, B, C for each group of constraints (as denoted by their indices) have consistent dimensions. Eq. (6.28) represents the objective function of the SGMSP; (6.29) are the linking constraints; (6.30) are other operational constraints; (6.31) are the constraints that map the maintenance variables y into z; (6.32), (6.33) are constraints that involve only decision variables y, z, respectively, and (6.34), (6.35), (6.36) define the domain of the decision variables. As shown in Chapter 5, fixing the coupling variables $z = \bar{z}$ in the linking constraints (6.29) splits the SGMSP into scenario-wise subproblems

$$Q_\omega(\bar{z}) = \max_{x_\omega} c^\intercal x_\omega \tag{6.37}$$

subject to

$$A_1 x_\omega \geq b_{1\omega} - C_1 \bar{z}, \tag{6.38}$$

$$A_2 x_\omega \geq b_{2\omega}, \tag{6.39}$$

$$x_\omega \geq 0, \tag{6.40}$$

and a master problem

$$\max_{y,z,w} \quad w - f^{\intercal} y \tag{6.41}$$

subject to

$$B_3 y + C_3 z = b_3 \tag{6.42}$$

$$B_4 y \geq b_4 \tag{6.43}$$

$$C_5 z \geq b_5 \tag{6.44}$$

$$w + C_6 z \geq b_6 \tag{6.45}$$

$$y \in \{0,1\}^{n_y} \tag{6.46}$$

$$z \in \{0,1\}^{n_z} \tag{6.47}$$

where w is the expected value of the subproblems corresponding to the solution z^*, as approximated by the optimality cuts iteratively generated by the Benders algorithm and included into the cut set (6.45). Due to constraints (6.42) which link the variables y and z, the master problem (6.41)-(6.47) is hard to solve for large instances. However, this problem can be greatly simplified by excluding from it the variables z and computing their values outside the model. With this simplification, the reduced master problem is

$$\max_{y,w} \quad w - f^{\intercal} y \tag{6.48}$$

subject to

$$B_4 y \geq b_4 \tag{6.49}$$

$$w + B_7 y \geq b_7 \tag{6.50}$$

$$w \geq 0 \tag{6.51}$$

$$y \in \{0,1\}^{n_y} \tag{6.52}$$

where (6.50) represents the set of cuts for approximating the expected solution value w of the subproblems (6.28)-(6.40) corresponding to the solution y^*. Unfortunately, as the decision variables of the master problem y are not directly fixed into the subproblems, the subproblem dual solutions cannot straightforwardly generate Benders optimality cuts for the reduced master problem. Instead we resort to Upper Bounding Functionals (UBF), as in Laporte and Louveaux (1993), for generating the cuts that approximate the values of w

based on trial solutions $\bar{y}$:

$$w \leq Q(\bar{y}) - (UB^{SP} - Q(\bar{y}))\Bigg(\sum_{(m,t)\in\mathcal{A}} y_{mt} - \sum_{(m,t)\notin\mathcal{A}} (y_{mt}) - |\mathcal{A}| \Bigg), \tag{6.53}$$

where $Q(\bar{y})$ denotes the expected objective value of the subproblems corresponding to the trial solution $\bar{y}$; UB^{SP} is the upper bound of the subproblems, and $\mathcal{A}$ is the set of variables in $\bar{y}$ with value equal to 1, i.e.,

$$\mathcal{A} = \{(m,t) \in \mathcal{M} \times \mathcal{T} \mid \bar{y}_{mt} = 1\}. \tag{6.54}$$

We refer to (6.53) as UBF cuts.

Proposition 5. *The UBF cuts* (6.53) *are valid, for any* $w \in \mathbb{R}$, $y \in \{0,1\}^{n_y}$, $\mathcal{A}$.

Proof. Notice that for any y and $\mathcal{A}$, the expression

$$\psi(y,\mathcal{A}) = \sum_{(m,t)\in\mathcal{A}} y_{mt} - \sum_{(m,t)\notin\mathcal{A}} (y_{mt}) - |\mathcal{A}|, \tag{6.55}$$

is non-positive, and its value is zero only if $y_{mt} = 1$, $\forall\,(m,t) \in \mathcal{A}$, and $y_{mt} = 0$, $\forall\,(m,t) \notin \mathcal{A}$. If $\psi(y,\mathcal{A}) = 0$, the right-hand side of (6.53) is equal to $Q(\bar{y})$, and the inequality $w \leq Q(\bar{y})$ is valid. If $\psi(y,\mathcal{A})$ is negative with absolute value K, (6.53) becomes

$$w \leq Q(\bar{y}) + K(UB^{SP} - Q(\bar{y})),$$

which can be rewritten as

$$w \leq UB^{SP} + (K-1)(UB^{SP} - Q(\bar{y})). \tag{6.56}$$

Since $UB^{SP} \geq Q(\bar{y})$ and $K \geq 0$, inequality (6.56) holds because its right-hand side is greater than UB^{SP}, and thus (6.53) is a valid cut. □

For a given master solution $\bar{y}$, the solution values of the variables z can be computed from the mapping constraints (5.8), i.e., from

$$r_{it} + \sum_{k\in\mathcal{K}(i,t)} k z_{itk} = \bar{G}_{it}, \quad \forall\,(i,t) \in \mathcal{I} \times \mathcal{T}, \tag{6.57}$$

by defining,

$$\bar{z}_{itk} = \begin{cases} 1 & \text{if} \quad k = \bar{G}_{it} - \bar{r}_{it} \\ 0 & \text{otherwise} \end{cases} \tag{6.58}$$

where, by (5.7),

$$\bar{r}_{it} = \sum_{\substack{m \in \mathcal{M}(i) \\ t' \in \mathcal{T}(m) \cap [t - D_m + 1, t]}} \bar{y}_{mt'}, \quad \forall \, (i,t) \in \mathcal{I} \times \mathcal{T}. \tag{6.59}$$

In summary, whereas in the decomposition method of Chapter 5, the Benders optimality cuts are computed from the dual solutions of the subproblems, and the master problem includes all the complicating variables (y and z) (Fig. 6.1), in the Alternative Decomposition Strategy (ADS) presented in this section, the master problem contains only the variables y, and the values of the variables z are computed outside the model (Fig. 6.2). Because in the ADS the dual solutions of the subproblems cannot directly generate Benders optimality cuts for the reduced master problem, we use UBF cuts (similar to Laporte and Louveaux, 1993) for approximating the expected optimal value of the subproblems, as defined in (6.53).

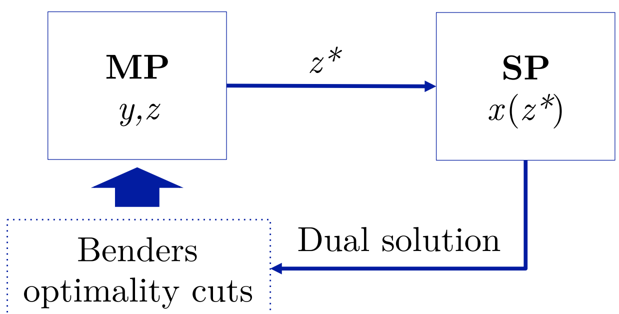


Figure 6.1 Schematic of the Benders decomposition method for the SGMSP with the master problem containing all the integer variables (y and z).

Although the reduced master problem (6.48)-(6.52) of the ADS can be solved faster than the Benders master problem of Chapter 5, the algorithm of the ADS may require a large number of iterations to converge for two reasons:

1. The reduced master problem contains less information about the complete problem, which can require more cuts for approximating the optimal solution.
2. The dual solutions of the subproblem are not exploited since the UBF cuts are computed using only the upper bounds and the objective values of the subproblems. Therefore, the UBF cuts are weaker than the Benders Optimality cuts of Chapter 5.

Therefore, computational experiments are necessary to compare the performance of the decomposition approaches discussed in this section. To test the ADS, we implemented the

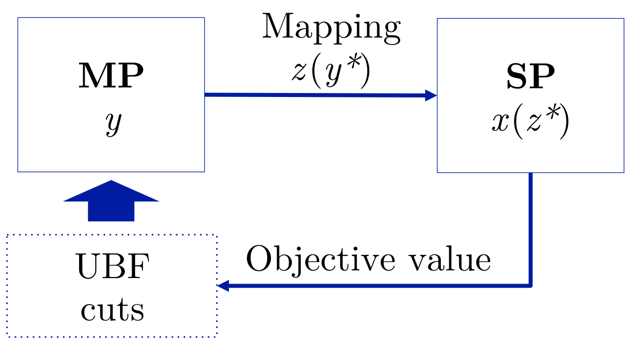


Figure 6.2 Schematic of the ADS' algorithm: the variables z are computed from the master problem solution y, and UBF cuts are generated from the subproblems' optimal values.

parallel algorithm shown in Fig. 6.3.

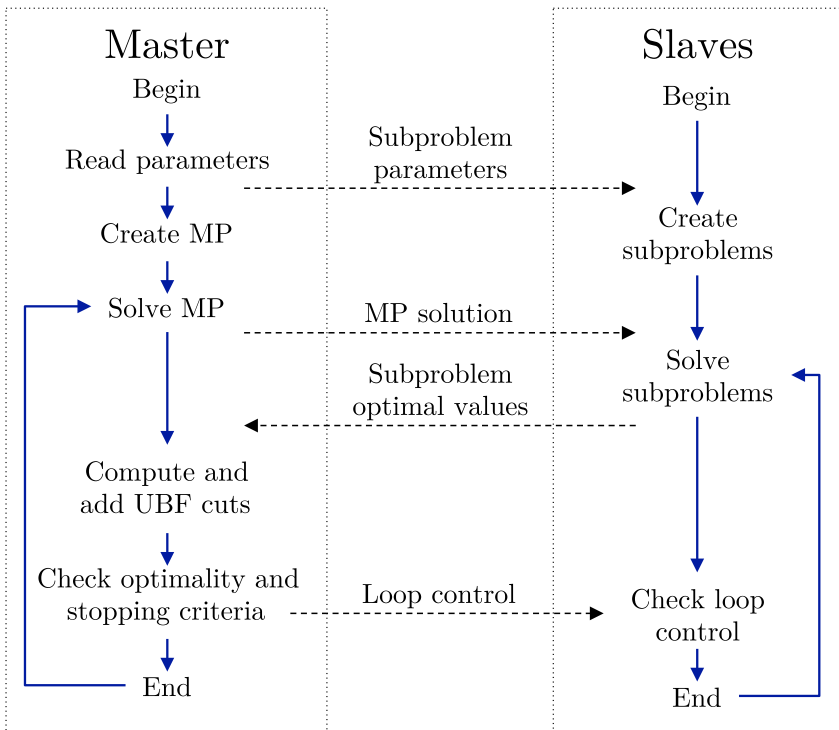


Figure 6.3 Sketch of the parallel algorithm based on the ADS (with a reduced master problem) implemented with MPI.

Because no other acceleration techniques were implemented for the parallel ADS algorithm, we compared this approach against the basic parallel Benders method of Chapter 5, i.e., with no acceleration techniques.

We ran tests on an 8-core desktop computer, with Xpress-MP for solving the master problem, Clp for solving the subproblems, and MPI for executing the parallelization. We used 20 instances with 20 time periods, and with up to 4 powerhouses, 12 maintenance tasks and 8 inflow scenarios. For each instance we defined a 1000 s time limit.

Our results indicate that the parallel ADS algorithm is faster than the Benders decomposition method only when the number of maintenance tasks is small (6 or less). When the number of tasks increases, the 1000 s time limit was reached in most of the instances, and the basic parallel Benders method achieved a smaller optimality gap than the optimality gap of the ADS algorithm. Naturally, the performance of the ADS and the Benders decomposition algorithms can improve significantly with acceleration techniques as those proposed in Chapter 5. In the next chapter we discuss further refinements to these methods.

Table 6.1 Computational time and relative optimality gap of parallel Benders and ADS methods on 20 instances, with 1000 s time limit. For each instance, the number of powerhouses, maintenance tasks and scenarios is indicated by the three digits of the column *Size*, in the corresponding order. The number of time periods in all instances is 20.

Instance		Time (s)		Rel. opt. gap	
Number	Size	ADS	Benders	ADS	Benders
1	2 - 4 - 8	8	20	-	-
2	4 - 6 - 4	186	442	-	-
3	4 - 8 - 4	-	-	0.084	0.060
4	4 - 10 - 4	.	-	0.116	0.040
5	4 - 12 - 4	-	-	0.158	0.078
6	2 - 4 - 8	8	12	-	-
7	4 - 6 - 4	117	190	-	-
8	4 - 8 - 4	-	-	0.016	0.012
9	4 - 10 - 4	-	-	0.154	0.052
10	4 - 12 - 4	-	-	0.203	0.100
11	2 - 4 - 8	11	17	-	-
12	4 - 6 - 4	154	297	-	-
13	4 - 8 - 4	-	-	0.017	0.015
14	4 - 10 - 4	245	554	-	-
15	4 - 12 - 4	-	-	0.056	0.022
16	2 - 4 - 8	7	10	-	-
17	4 - 6 - 4	180	515	-	-
18	4 - 8 - 4	-	-	0.022	0.016
19	4 - 10 - 4	-	-	0.152	0.051
20	4 - 12 - 4	-	-	0.250	0.117

CHAPTER 7 GENERAL DISCUSSION

As shown in the previous chapters, modelling the techno-economic characteristics of the hydroelectricity production within the GMSP leads to a challenging nonlinear stochastic combinatorial optimization problem, namely, the SGMSP. This section summarizes our MILP-based solution approaches to this problem, discusses their limitations and outlines future research.

7.1 Synthesis of work

Tables 7.1-7.3 summarize the research outcomes of this project, according to the objectives defined in Section 1.3. Next, we discuss our contributions to the SGMSP in hydropower systems, according to the two main topics of this study:

1. MILP formulations for SGMSP (Objectives 1 and 4).
2. Decomposition methods for SGMSP (Objectives 2, 3 and 4).

Table 7.1 Summary of contributions I

Objectives	Approach	Outcomes	Sections	Highlights
1. To develop a tightened mixed-integer programming formulation for the GMSP, considering the time windows of maintenance activities and the nonlinearities of the hydropower production function.	Mixed-integer linear programming	Basic formulation for GMSP in hydropower systems	4.3.2	• Using hyperplanes for approximating the nonlinarities of hydroelectricity production.
			4.3.3	• Compact formulation: using index sets to avoid explicit definition of time window constraints and to exclude other unncessary variables and constraints.
			4.3.5, 4.8.1	• We prove that under some conditions, in any feasible solution obtained with a general MILP solver, there is no simultaneous purchase and sale of electricity.
	Tightening techniques for MILP	Set reduction technique	4.4.2	• We exploit time windows information to reduce the model size.
		Extended formulation	4.4.1	• Including additional variables and constraints for tighening the formulation.
		Four families of valid inequalities	4.4.3, 5.8.1	• We prove that under some conditions, the proposed valid inequalities allow relaxing the integrality conditions.
	Computational experiments	Selection of best combination of tighening techniques for GMSP in hydropower systems	4.5	• We propose a normalization method for comparing computational times of different instances. • The extended formulation had the most significant effect on reducing computational times. • Set reduction in combination with extended formulation had the best average performance. • Best formulation was about 10 times faster than the basic formulation.
		Industrial application example	4.6	• We show that neglecting nonlinearities of hydroelectricity production significantly overestimates the electricity production and leads to suboptimal decisions.
	Stochastic programming	Two-stage stochastic linear program for GMSP	5.2	• Maintenance decisions in the first stage and second stage with scenario fan of water inflows.

Table 7.2 Summary of contributions II

Objectives	Approach	Outcomes	Sections	Highlights
2. To implement a Benders decomposition method for the SGMSP with uncertain water inflows.	Benders decomposition	Implementation of Benders decomposition for SGMSP in hydropower systems	5.3.2 5.8.1	• Decomposition with all binary variables in master problem. • Using reduced costs for computing contributions of variable bounds. • We derive necessary conditions for relatively complete recourse (feasibility cuts are unnecessary).
3. To accelerate the Benders decomposition method for the SGMSP by means of parallelization and acceleration techniques.	Parallel programming	Parallelization of Benders decomposition	5.4.2	• Using C++ with the MPI protocol, and Xpress BCL. Solving the master problem with Xpress-MP and the subproblems with open-source solver CLP.
	Acceleration techniques for Benders decomposition	Valid inequalities, warm start, multi-phase relaxation, SOS, combinatorial cuts, presolve and integer rounding cuts	5.4.1 5.4.2	• New acceleration techniques: warm start, special ordered sets, presolve. • We derive new combinatorial cuts, and we show how to apply them for removing non-optimal solutions. • Exploiting funcionalities of Xpress, such as delayed rows for optimality cuts and model cuts for valid inequalities.
	Computational experiments	Selecting the best combination of acceleration techniques	5.5.1	• A combination of five techniques had the lowest computational time: presolve, special ordered sets, warm start, combinatorial cuts and integer rounding cuts.
				• Fourfold speedup using the best combination of acceleration techniques.
		Methodology for efficiently selecting combination of techniques	5.5.1	• Applying statistical methods for sequential experiments.
		Testing effect of parallelization	5.5.2	• Parallelized Benders ouperforms direct MILP solution above 50-100 inflow scenarios.
				• Solution to optimality in instances with 4 powerhouses, 15 time periods, 8 tasks and 200 scenarios.

Table 7.3 Summary of contributions III

Objectives	Approach	Outcomes	Sections	Highlights
4. To outline extensions to the mathematical program for the SGMSP and to the solution approach.	Model extensions: mixed-integer linear programming	Additional constraints: resourse utilization, mutually exclusive tasks, overlapping of tasks, precedence of starting times and execution	6.1.1	• Compact formulation using index sets.
		Formulation with discrete choice of durations of tasks	6.1.2	• Compact formulation using index sets.
	Optimization under uncertainty of load	Chance-constrained formulation	6.1.3	• Including auxiliary variables and constraints for operation under extreme load.
		Max-min formulation	6.1.3	• Maximizing the minimum net reserve.
	Alternative decomposition strategy (ADS)	Using a reduced master problem and UBF cuts	6.2	• Including only a subset of complicating variables in master problem. • Smaller master problem but dual solution of subproblems not exploited. • Parallelized implementation of ADS with MPI • Low scalability of ADS on number of maintenance tasks, in comparison with basic parallel Benders

7.1.1 MILP formulations for SGMSP

This project was motivated by the need for developing a solution method for maintenance scheduling of generating units in the hydropower system of Rio Tinto Aluminium. The company demanded a scheduling method with an accurate representation of the uncertain water inflows and the nonlinearity of the hydroelectricity production. In the literature, this problem was virtually unexplored.

Our MILP formulation for the GMSP is the first one to approximate the hydroelectricity production as a nonlinear function of water discharges, stored water levels and number of active generators. As discussed in Section 2.2.3, Helseth et al. (2018) represented only the nonlinearity of turbine discharges, and Ge et al. (2018) and Guedes et al. (2015) considered the water discharges and stored water levels, but neglected the nonlinear effect of the number of active generators.

To represent the nonlinearity of the electricity production on the number of active generators, our MILP formulation includes:

1. Approximating hyperplanes of the hydroelectricity production function for each number of active generators in each powerhouse (Section 4.3.2).
2. Constraints for mapping the maintenance schedule variables y into the indicator variables of the number of active generators z (Section 4.3.3).

As these elements lead to a mathematical program with a difficult structure and a poor continuous relaxation, first, we thinned out our formulation using three simple ideas:

1. We exclude from the model superfluous elements by defining conditions on the combinations of the index sets of variables and constraints. For example, we avoid defining non-binding constraints and variables with a fixed value.
2. The time windows of activities are implicitly defined by the decision variables and not by general constraints (see Dahal et al., 2015, for a similar modelling approach),
3. Whenever possible, we restrict the domain of the variables using variable bounds instead of general constraints.

Second, we explored three tightening techniques: valid inequalities (Sections 4.4.3, 5.8.1), disaggregation of variables and constraints (referred to as *extended formulation* in Section 4.4.1), and reduction of the model elements based on the time window parameters (referred to as *set reduction* in Section 4.4.2). In computational experiments with a MILP solver, the extended formulation in combination with set reduction was solved up to 10 times faster than the basic formulation (Section 4.5).

To represent the uncertain water inflows, we extended the formulation of Chapter 4 as a two-stage stochastic program, with maintenance scheduling decisions in the first stage, and scenario-wise operational decisions in the second stage (Section 5.2.2). In hydropower maintenance scheduling, the uncertainty of water inflows was only addressed recently. Ge et al. (2018) proposed a chance-constrained approach, and Helseth et al. (2018), considering a simpler scheduling problem, used a multi-stage model of the hydropower operation (see Section 2.2.3 for a discussion).

Numerical experiments showed that both the number of inflow scenarios and the nonlinearity of the hydroelectricity production function have a significant impact on the estimated objective value of the solutions (see Sections 4.6, 5.5.2). In our computational experience with the proposed formulation, a commercial MILP solver achieved optimal and near-optimal solutions in a reasonable times (Section 4.5), when applied to small and mid-size instances of the deterministic problem.

Furthermore, we showed that the compact formulation of Chapters 4 and 5 can be straightforwardly extended to select the duration of activities (Section 6.1.2) and to define constraints of precedence, overlapping, non-simultaneous activities and available maintenance, using index set conditions (Section 6.1.1). Similar constraints have been considered in previous works, but with a modelling approach that adds significant complexity to the problem (see for example Perez-Canto and Rubio-Romero, 2013).

Finally, we outlined chance-constrained and min-max formulations for ensuring generation reserves to buffer the forecast errors of the electricity load (Section 6.1.3).

7.1.2 Decomposition methods for SGMSP

Benders decomposition with acceleration techniques

We implemented seven techniques for speeding up our Benders decomposition method for the SGMSP (Section 5.4.1):

— Valid inequalities
— Warm start
— Multi-phase relaxation
— Special ordered sets
— Combinatorial cuts
— Presolve
— Integer rounding cuts

To the best of our knowledge, we present an original approach for applying presolve, warm start and special ordered sets to Benders decomposition. In presolve, we reduce the Benders master problem using presolve information from the complete problem. In special ordered sets, we define branching directives and ordering of variables in the master problem, based on subproblem parameters. In warm start, we speed up the master problem solution using bounds, cutoff values and initial solutions extracted from the previous Benders iteration. Furthermore, we developed cuts that dominate the combinatorial cuts of Codato and Fischetti (2006), and we show how to apply such cuts for accelerating the Benders algorithm (Section 5.4.1).

For efficiently selecting the combination of acceleration techniques with the best impact on the computational time of the Benders algorithm, we applied statistical methods for defining experimental conditions, normalizing computational times, assessing the statistical significance of experimental factors, and running sequential experiments (see Sections 5.5 and 5.8.2).

Our computational experiments showed that, among the seven acceleration techniques, presolve had the single largest impact, and the combination presolve, special ordered sets, warm start, combinatorial cuts and integer rounding cuts yielded the largest speed up in our implementation of the Benders algorithm.

Furthermore, since warm start and presolve are problem-independent, such techniques can be included in standard implementations of Benders decomposition with a sequential solution of the master problem.

Due to the definition of variable bounds in our compact MILP formulation, we used values of the reduced costs to compute the marginal contribution of the binding variable bounds.

Parallelization and implementation of the Benders decomposition method

The available computational resources at the company significantly determined our approach for implementing the Benders decomposition method. For example, we used the open-source solver Clp for solving a large number of parallel subproblems on the cluster of the company, without license restrictions. For solving the master problem, we exploited the capabilities of the commercial solver Xpress-MP and the modelling libraries Xpress BCL. We observed that appropriately applying functionalities such as multi-threading, SOS, presolve, delayed rows (for Benders optimality cuts), and model cuts (for defining valid inequalities), significantly reduced the computational time of the decomposition algorithm. However, the interaction of Clp and Xpress-MP in our implementation of the Benders algorithm posed a challenge to its parallelization, due to different model representations in each solver.

Our computational tests on the computer cluster confirmed the high scalability of the parallelized Benders algorithm on the number of scenarios, and showed a significant reduction of the variance in the optimal values as the number of scenarios increases (Section 5.5.2).

Alternative decomposition strategy

Because the Benders master problem of Chapter 5 has a complicated mathematical structure, in Section 6.2 we explored an alternative decomposition strategy with a simplified master problem. However, in our computational experiments the scalability of this strategy on number of maintenance tasks was not competitive with the performance of the Benders algorithm (Section 6.2).

7.2 Study limitations and future research

7.2.1 Model extensions

Although Section 6.1.3 introduced approaches for improving the reliability of the solutions to the SGMSP, the trade-offs between cost and reliability of maintenance schedules can be more adequately assessed through multi-objective approaches, which can be subject of future research.

Future works can also incorporate other relevant aspects of the problem, such as transmission system constraints, uncertain duration of maintenance activities, and marginal values of water. For example, the marginal value of water can be considered by including into the objective function a term $\sum_{i \in \mathcal{I}} A_i s_{iT}$, where A_i is the marginal value of stored water in the reservoir of powerhouse i at the end of the planning horizon, in [\$/hm^3], and s_{iT} is the final stored water level at i, in [hm^3]. The parameter A_i can be estimated using mid-term models of the hydropower operation (Philpott, 2017).

For representing the non-anticipative decisions of the hydropower operation, future approaches can explore the impact of multi-stage (as in Helseth et al., 2018), with a better approximation of the nonlinear hydroelectricity production function (as in Section 4.3.2), and with a more realistic representation of the maintenance scheduling problem (Sections 4.3.3, 6.1).

Furthermore, we recommend exploring extensions and applications of the compact formulation to scheduling problems with similar characteristics, such as discrete-time scheduling of chemical processes (Floudas and Lin, 2004) and resource-constrained project scheduling (Kreter et al., 2016).

7.2.2 Refinements to the implemented solution methods

Due to the nontrivial structure of the SGMSP, the computational times of the explored solution approaches tend to rapidly increase with the number of maintenance tasks.

To speed up the solution of the SGMSP, a promising work consists in implementing a branch-and-Benders-cut method (Fortz and Poss, 2009), based on the Benders partitioning approach of Section 5. Also, the alternative decomposition strategy of Chapter 6.2 can be implemented in a branch and cut framework, using an integer L-shaped method (Laporte and Louveaux, 1993; Angulo et al., 2016). Naturally, acceleration techniques for such decomposition approaches would also be necessary.

Furthermore, the problem size can be reduced by aggregating water inflow scenarios through time series clustering (Liao, 2005).

7.2.3 Sub-decomposition approach

In the alternative decomposition strategy of Section 6.2 the gains of solving a reduced master problem (with only the scheduling variables y) were counteracted by the lack of dual solution information in the generated UBF cuts. For solving a reduced master problem with Benders optimality cuts, we recommend the decomposition strategy outlined in Fig. 7.1.

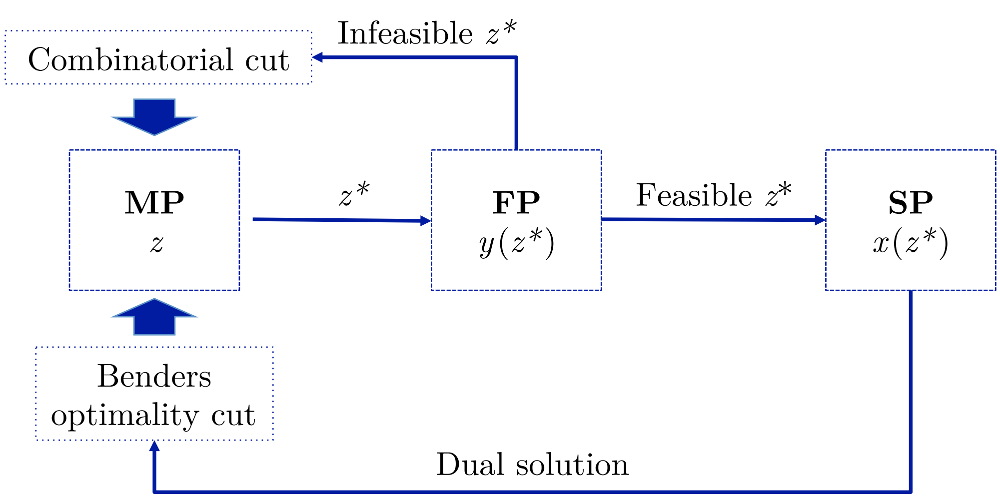


Figure 7.1 Outline of sub-decomposition approach

In this decomposition strategy, the original problem is split into a set of subproblems SP with the operational decisions x, a feasibility problem FP with the scheduling variables y, and a master problem MP with the binary variables z that indicate the number of active generators. Because the decision variables of MP are fixed into the subproblems SP, their dual solutions can be used to compute Benders optimality cuts. For any given MP solution z^*, the FP checks if there is a feasible schedule $y^*(z^*)$. If the MP solution z^* is feasible for FP, the subproblems SP are solved with the fixed variables $z = z^*$, and a Benders optimality cut is generated using the dual solution of the subproblems. If the MP solution z^* is not feasible, a combinatorial Benders cut is included into MP to remove this solution.

Notice that in the decomposition approach of Fig. 7.1, we assume costs of activities independent of the starting period. If such costs are indexed in time, FP must be replaced by an optimization problem that finds the minimum-cost maintenance schedule y^*, for a given MP solution z^*.

CHAPTER 8 CONCLUSION AND RECOMMENDATIONS

Motivated by a real problem in industry, we developed the first realistic MILP formulation for hydropower generator maintenance scheduling in hydropower systems. Our model represents relevant elements of the problem, such as the uncertainty of the water inflows, the time windows of maintenance activities and the nonlinearity of the hydroelectricity production. This formulation can accurately approximate the nonlinear effect of the maintenance outages on the hydropower operation.

Because the resulting MILP is hard to solve, we thinned out and tightened our model using a combination of techniques, which significantly reduced the computational time for solving the problem with a MILP solver. This formulation can be extended to represent more general constraints and decisions in maintenance scheduling.

Considering the significant uncertainty and variability of the water inflows, we implemented a parallel Benders decomposition method for solving the problem with a large number of inflow scenarios. A combination of acceleration techniques significantly reduced the computational times of our Benders implementation. Two of the novel acceleration techniques that we proposed are problem-independent and thus can be included in standard implementations of the Benders algorithm.

Due to the complicated structure of the master problem in the Benders decomposition approach, we implemented an alternative decomposition strategy with a reduced master problem. However, in computational experiments, the Benders decomposition method was more competitive than the alternative strategy.

The decomposition methods developed in this dissertation can be refined in future works through additional acceleration techniques and branch-and-cut implementations. Solution methods with a sub-decomposition of the master problem, can also be subject of future research. We outline this approach in Section 7.2.3. Future works can also explore applications of our compact modelling approach to similar scheduling problems, such as resource-constrained project scheduling and discrete-time scheduling of chemical processes.

Although the specific needs of the company drove this research, and the approach in this project was subject to the available information and computational resources, our experience confirms that collaboration with industry conveys opportunities to connect theory and practice, and to bring new challenges for research in academia.

REFERENCES

R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, "Network flows," 1988.

E. Álvarez-Miranda, I. Ljubić, M. Luipersbeck, and M. Sinnl, "Solving minimum-cost shared arborescence problems," *European Journal of Operational Research*, vol. 258, no. 3, pp. 887–901, 2017.

G. Angulo, S. Ahmed, and S. S. Dey, "Improving the integer L-shaped method," *INFORMS Journal on Computing*, vol. 28, no. 3, pp. 483–499, Aug. 2016. [Online]. Available: https://doi.org/10.1287/ijoc.2016.0695

A. Arce, "Optimal dispatch of generating units of the Itaipu hydroelectric plant," *IEEE Power Engineering Review*, vol. 21, no. 11, pp. 56–56, 2001.

C. Barnhart, E. L. Johnson, G. L. Nemhauser, M. W. Savelsbergh, and P. H. Vance, "Branch-and-price: Column generation for solving huge integer programs," *Operations Research*, vol. 46, no. 3, pp. 316–329, 1998.

M. T. Barros, F. T. Tsai, S.-l. Yang, J. E. Lopes, and W. W. Yeh, "Optimization of large-scale hydropower system operations," *Journal of Water Resources Planning and Management*, vol. 129, no. 3, pp. 178–188, 2003.

E. M. L. Beale and J. A. Tomlin, "Special facilities in a general mathematical programming system for non-convex problems using ordered sets of variables," in *Proc. of the 5th Int. Conf. on Operations Research*, J. R. Lawrence, Ed. Tavistock Publications, 1970, pp. 447–454.

J. F. Benders, "Partitioning procedures for solving mixed-variables programming problems," *Numerische Mathematik*, vol. 4, no. 1, pp. 238–252, 1962.

D. P. Bertsekas, *Dynamic programming and optimal control.* Athena scientific Belmont, MA, 1995, vol. 1, no. 2.

B. Bezerra, Á. Veiga, L. A. Barroso, and M. Pereira, "Stochastic long-term hydrothermal scheduling with parameter uncertainty in autoregressive streamflow models," *IEEE Transactions on Power Systems*, vol. 32, no. 2, pp. 999–1006, 2017.

R. Billinton and R. N. Allan, *Reliability evaluation of power systems.* Springer Science & Business Media, 1996.

J. R. Birge and F. Louveaux, *Introduction to stochastic programming.* Springer Science & Business Media, 2011.

J. R. Birge and F. V. Louveaux, “A multicut algorithm for two-stage stochastic linear programs,” *European Journal of Operational Research*, vol. 34, no. 3, pp. 384–392, 1988.

E. R. Bixby, M. Fenelon, Z. Gu, E. Rothberg, and R. Wunderling, “MIP: Theory and practice-—closing the gap,” in *IFIP Conference on System Modeling and Optimization*. Springer, 1999, pp. 19–49.

R. Bixby and E. Rothberg, “Progress in computational mixed integer programming—a look back from the other side of the tipping point,” *Annals of Operations Research*, vol. 149, no. 1, pp. 37–41, 2007.

N. L. Boland and A. C. Eberhard, “On the augmented Lagrangian dual for integer programming,” *Mathematical Programming*, vol. 150, no. 2, pp. 491–509, 2015.

A. Borghetti, C. D’Ambrosio, A. Lodi, and S. Martello, “An MILP approach for short-term hydro scheduling and unit commitment with head-dependent reservoir,” *IEEE Transactions on Power Systems*, vol. 23, no. 3, pp. 1115–1124, 2008.

Q. Botton, B. Fortz, L. Gouveia, and M. Poss, “Benders decomposition for the hop-constrained survivable network design problem,” *INFORMS Journal on Computing*, vol. 25, no. 1, pp. 13–26, 2013.

S. Bradley, A. Hax, and T. Magnanti, *Applied Mathematical Programming*. Addison-Wesley Publishing Company, 1977. [Online]. Available: https://books.google.ca/books?id=MSWdWv3Gn5cC

G. Budai, R. Dekker, and R. P. Nicolai, “Maintenance and production: a review of planning models,” in *Complex system maintenance handbook*. Springer, 2008, pp. 321–344.

C. C. Carøe and J. Tind, “L-shaped decomposition of two-stage stochastic programs with integer recourse,” *Mathematical Programming*, vol. 83, no. 1-3, pp. 451–464, 1998.

P.-L. Carpentier, M. Gendreau, and F. Bastin, “Long-term management of a hydroelectric multireservoir system under uncertainty using the progressive hedging algorithm,” *Water Resources Research*, vol. 49, no. 5, pp. 2812–2827, 2013.

J. Catalão, S. Mariano, V. Mendes, and L. Ferreira, “Scheduling of head-sensitive cascaded hydro systems: A nonlinear approach,” *IEEE Transactions on Power Systems*, vol. 24, no. 1, pp. 337–346, 2009.

D. Chattopadhyay, K. Bhattacharya, and J. Parikh, “A systems approach to least-cost maintenance scheduling for an interconnected power system,” *IEEE Transactions on Power Systems*, vol. 10, no. 4, pp. 2002–2007, 1995.

D.-S. Chen, R. G. Batson, and Y. Dang, *Applied integer programming: modeling and solution*. John Wiley & Sons, 2011.

G. Codato and M. Fischetti, “Combinatorial Benders’ cuts for mixed-integer linear programming,” *Operations Research*, vol. 54, no. 4, pp. 756–766, 2006.

A. J. Conejo, J. M. Arroyo, J. Contreras, and F. A. Villamor, “Self-scheduling of a hydro producer in a pool-based electricity market,” *IEEE Transactions on Power Systems*, vol. 17, no. 4, pp. 1265–1272, 2002.

A. J. Conejo, R. García-Bertrand, and M. Díaz-Salazar, “Generation maintenance scheduling in restructured power systems,” *IEEE Transactions on Power Systems*, vol. 20, no. 2, pp. 984–992, 2005.

J.-F. Cordeau, G. Stojković, F. Soumis, and J. Desrosiers, “Benders decomposition for simultaneous aircraft routing and crew scheduling,” *Transportation Science*, vol. 35, no. 4, pp. 375–388, 2001.

J.-F. Cordeau, F. Pasin, and M. M. Solomon, “An integrated model for logistics network design,” *Annals of Operations Research*, vol. 144, no. 1, pp. 59–82, 2006.

J.-F. Cordeau, F. Furini, and I. Ljubić, “Benders decomposition for very large scale partial set covering and maximal covering problems,” 2018. [Online]. Available: http://www.optimization-online.org/DB_FILE/2018/06/6665.pdf

M. Cordova, E. Finardi, F. Ribas, V. de Matos, and M. Scuzziato, “Performance evaluation and energy production optimization in the real-time operation of hydropower plants,” *Electric Power Systems Research*, vol. 116, pp. 201–207, 2014.

A. M. Costa, “A survey on Benders decomposition applied to fixed-charge network design problems,” *Computers & Operations Research*, vol. 32, no. 6, pp. 1429–1450, 2005.

T. G. Crainic, M. Hewitt, and W. Rei, “Partial Benders decomposition strategies for two-stage stochastic integer programs,” CIRRELT, Tech. Rep. CIRRELT-2016-37, 2016.

P. Côté, C. Audet, N. Amaioua, E. Bigeon, Q. Desreumaux, A. Ihaddadene, Y. Mir, J. A. Rodríguez Sarasty, and L. Zéphyr, “Planning of the maintenance outages for a set of hydroelectric turbogenerators,” in *Proceedings of the Sixth Montreal Industrial Problem Solving Workshop*, O. Marcotte, Ed. CRM Research Report CRM-3350, 2015, pp. 1–10.

K. Dahal, K. Al-Arfaj, and K. Paudyal, “Modelling generator maintenance scheduling costs in deregulated power markets,” *European Journal of Operational Research*, vol. 240, no. 2, pp. 551–561, 2015.

G. B. Dantzig, “Linear programming,” *Operations Research*, vol. 50, no. 1, pp. 42–47, 2002.

G. B. Dantzig and P. Wolfe, “The decomposition algorithm for linear programs,” *Econometrica: Journal of the Econometric Society*, pp. 767–778, 1961.

R. Dekker, “Applications of maintenance optimization models: a review and analysis,” *Reliability Engineering & System Safety*, vol. 51, no. 3, pp. 229–240, 1996.

G. Desaulniers, J. Desrosiers, M. M. Solomon, F. Soumis, D. Villeneuve *et al.*, “A unified framework for deterministic time constrained vehicle routing and crew scheduling problems,” in *Fleet management and logistics.* Springer, 1998, pp. 57–93.

G. Desaulniers, J. Desrosiers, and M. M. Solomon, *Column generation.* Springer Science & Business Media, 2006, vol. 5.

A. L. Diniz and M. E. P. Maceira, “A four-dimensional model of hydro generation for the short-term hydrothermal dispatch problem considering head and spillage effects,” *IEEE Transactions on Power Systems*, vol. 23, no. 3, pp. 1298–1308, 2008.

E. D. Dolan and J. J. Moré, “Benchmarking optimization software with performance profiles,” *Mathematical Programming*, vol. 91, no. 2, pp. 201–213, 2002.

Y. Feng, P. Li, and H. Wang, “Hydro-thermal generator maintenance scheduling accommodating both randomness and fuzziness,” in *Electric Utility Deregulation and Restructuring and Power Technologies (DRPT), 2011 4th International Conference on.* IEEE, 2011, pp. 734–741.

Fichtner, *Hydroelectric Power: A Guide for Developers and Investors*, International Finance Corporation, 2015.

FICO, *Xpress-Optimizer reference manual*, 2017.

——, *Xpress-BCL reference manual*, 2014.

E. C. Finardi and E. L. da Silva, “Solving the hydro unit commitment problem via dual decomposition and sequential quadratic programming,” *IEEE Transactions on Power Systems*, vol. 21, no. 2, pp. 835–844, 2006.

M. Fischetti, I. Ljubić, and M. Sinnl, “Benders decomposition without separability: A computational study for capacitated facility location problems,” *European Journal of Operational Research*, vol. 253, no. 3, pp. 557–569, 2016.

——, “Redesigning Benders decomposition for large-scale facility location,” *Management Science*, vol. 63, no. 7, pp. 2146–2162, 2016.

C. A. Floudas and X. Lin, “Continuous-time versus discrete-time approaches for scheduling of chemical processes: a review,” *Computers & Chemical Engineering*, vol. 28, no. 11, pp. 2109–2129, 2004.

W. K. Foong, A. R. Simpson, H. R. Maier, and S. Stolp, “Ant colony optimization for power plant maintenance scheduling optimization—– a five-station hydropower system,” *Annals of Operations Research*, vol. 159, no. 1, pp. 433–450, 2008.

B. Fortz and M. Poss, "An improved Benders decomposition applied to a multi-layer network design problem," *Operations Research Letters*, vol. 37, no. 5, pp. 359–364, 2009.

A. Froger, M. Gendreau, J. E. Mendoza, É. Pinson, and L.-M. Rousseau, "Maintenance scheduling in the electricity industry: A literature review," *European Journal of Operational Research*, vol. 251, no. 3, pp. 695–706, 2016.

K. Fukuda. (2011) Polyhedral computation software. Last checked: 2017-02-15. [Online]. Available: https://www.inf.ethz.ch/personal/fukudak/soft/soft.html

C. Gauvin, E. Delage, and M. Gendreau, "Decision rule approximations for the risk averse reservoir management problem," *European Journal of Operational Research*, vol. 261, no. 1, pp. 317–336, 2017.

X.-l. Ge, L.-z. Zhang, J. Shu, and N.-f. Xu, "Short-term hydropower optimal scheduling considering the optimization of water time delay," *Electric Power Systems Research*, vol. 110, pp. 188–197, 2014.

X. Ge, S. Xia, and X. Su, "Mid-term integrated generation and maintenance scheduling for wind-hydro-thermal systems," *International Transactions on Electrical Energy Systems*, 2018.

B. Gendron, M. G. Scutellà, R. G. Garroppo, G. Nencioni, and L. Tavanti, "A branch-and-Benders-cut method for nonlinear power design in green wireless local area networks," *European Journal of Operational Research*, vol. 255, no. 1, pp. 151–162, 2016.

A. M. Geoffrion, "Lagrangian relaxation for integer programming," in *50 Years of Integer Programming 1958-2008*. Springer, 2010, pp. 243–281.

B. Gorenstin, N. Campodonico, J. Costa, and M. Pereira, "Power system expansion planning under uncertainty," *IEEE Transactions on Power Systems*, vol. 8, no. 1, pp. 129–136, 1993.

L. Guedes, D. Vieira, A. Lisboa, and R. Saldanha, "A continuous compact model for cascaded hydro-power generation and preventive maintenance scheduling," *International Journal of Electrical Power & Energy Systems*, vol. 73, pp. 702–710, 2015.

M. Guignard and S. Kim, "Lagrangean decomposition: A model yielding stronger lagrangean bounds," *Mathematical Programming*, vol. 39, no. 2, pp. 215–228, 1987.

A. Helseth, M. Fodstad, and B. Mo, "Optimal hydropower maintenance scheduling in liberalized markets," *IEEE Transactions on Power Systems*, 2018.

R. Henrion, "Introduction to chance-constrained programming," 2004. [Online]. Available: http://www.wias-berlin.de/people/henrion/ccp.ps

J. N. Hooker and G. Ottosson, "Logic-based benders decomposition," *Mathematical Programming*, vol. 96, no. 1, pp. 33–60, 2003.

M. T. Jonsson, "Power plant maintenance scheduling using dependency structure matrix and evolutionary optimization," in *Proceedings of the World Congress on Engineering and Computer Science 2015*, editor, Ed., 2015.

E. Klotz and A. M. Newman, "Practical guidelines for solving difficult mixed integer linear programs," *Surveys in Operations Research and Management Science*, vol. 18, no. 1-2, pp. 18–32, 2013.

S. Kreter, J. Rieck, and J. Zimmermann, "Models and solution procedures for the resource-constrained project scheduling problem with general temporal constraints and calendars," *European Journal of Operational Research*, vol. 251, no. 2, pp. 387–403, 2016.

I. Kuzle, H. Pandžić, and M. Brezovec, "Hydro generating units maintenance scheduling using Benders decomposition," *Tehnički vjesnik*, vol. 17, no. 2, pp. 145–152, 2010.

A. H. Land and A. G. Doig, "An automatic method of solving discrete programming problems," *Econometrica: Journal of the Econometric Society*, pp. 497–520, 1960.

G. Laporte and F. V. Louveaux, "The integer L-shaped method for stochastic integer programs with complete recourse," *Operations Research Letters*, vol. 13, no. 3, pp. 133–142, 1993.

L. S. Lasdon, *Optimization theory for large systems.* Courier Corporation, 1970.

M. Leitner, I. Ljubić, M. Luipersbeck, and M. Sinnl, "Decomposition methods for the two-stage stochastic Steiner tree problem," *Computational Optimization and Applications*, vol. 69, no. 3, pp. 713–752, 2018.

T. W. Liao, "Clustering of time series data—a survey," *Pattern Recognition*, vol. 38, no. 11, pp. 1857–1874, 2005.

J. Linderoth and S. Wright, "Decomposition algorithms for stochastic programming on a computational grid," *Computational Optimization and Applications*, vol. 24, no. 2-3, pp. 207–250, 2003.

I. Ljubić, P. Mutzel, and B. Zey, "Stochastic survivable network design problems: Theory and practice," *European Journal of Operational Research*, vol. 256, no. 2, pp. 333–348, 2017.

T. L. Magnanti and R. T. Wong, "Accelerating Benders decomposition: Algorithmic enhancement and model selection criteria," *Operations Research*, vol. 29, no. 3, pp. 464–484, 1981.

A. Marchand, M. Gendreau, M. Blais, and G. Emiel, "Fast near-optimal heuristic for the short-term hydro-generation planning problem," *IEEE Transactions on Power Systems*, vol. 33, no. 1, pp. 227–235, 2018.

D. McDaniel and M. Devine, "A modified Benders' partitioning algorithm for mixed integer programming," *Management Science*, vol. 24, no. 3, pp. 312–319, 1977.

D. C. Montgomery and G. C. Runger, *Applied statistics and probability for engineers*. John Wiley & Sons, 2010.

L. M. Moro and A. Ramos, "Goal programming approach to maintenance scheduling of generating units in large scale power systems," *IEEE Transactions on Power Systems*, vol. 14, no. 3, pp. 1021–1028, 1999.

G. L. Nemhauser and L. A. Wolsey, *Integer and combinatorial optimization*. New York, NY, USA: Wiley-Interscience, 1988.

S. S. Nielsen and S. A. Zenios, "Scalable parallel Benders decomposition for stochastic linear programming," *Parallel Computing*, vol. 23, no. 8, pp. 1069–1088, 1997.

G. Oliveira, S. Granville, and M. Pereira, "Optimization in electrical power systems," in *Handbook of applied optimization*, P. Pardalos and M. Resende, Eds. Oxford University Press, 2002, ch. 18.8.1, p. 770–807.

N. Papadakos, "Practical enhancements to the Magnanti–Wong method," *Operations Research Letters*, vol. 36, no. 4, pp. 444–449, 2008.

G. Papazafeiropoulos. (2014) Computational geometry toolbox. [Online]. Available: https://goo.gl/TYuv3E

M. V. Pereira and L. M. Pinto, "Multi-stage stochastic optimization applied to energy planning," *Mathematical Programming*, vol. 52, no. 1-3, pp. 359–375, 1991.

S. Perez-Canto, "Application of Benders' decomposition to power plant preventive maintenance scheduling," *European Journal of Operational Research*, vol. 184, no. 2, pp. 759–777, 2008.

S. Perez-Canto and J. C. Rubio-Romero, "A model for the preventive maintenance scheduling of power plants including wind farms," *Reliability Engineering & System Safety*, vol. 119, pp. 67–75, 2013.

A. Philpott, "On the marginal value of water for hydroelectricity," in *Advances and Trends in Optimization with Engineering Applications*, M. F. A. Tamás Terlaky, D. C. Mowery, and S. Ahmed, Eds. SIAM, 2017, ch. 31, pp. 405–425.

R. Rahmaniani, T. G. Crainic, M. Gendreau, and W. Rei, "The Benders decomposition algorithm: A literature review," *European Journal of Operational Research*, vol. 259, no. 3, pp. 801–817, 2017.

T. Ralphs, Y. Shinano, T. Berthold, and T. Koch, "Parallel solvers for mixed integer linear optimization," in *Handbook of parallel constraint reasoning*. Springer, 2018, pp. 283–336.

R. T. Rockafellar and R. J.-B. Wets, "Scenarios and policy aggregation in optimization under uncertainty," *Mathematics of Operations Research*, vol. 16, no. 1, pp. 119–147, 1991.

J. A. Rodríguez, M. F. Anjos, P. Côté, and G. Desaulniers, "MILP formulations for generator maintenance scheduling in hydropower systems," Les Cahiers du GERAD, Tech. Rep. G-2017-63, August 2017.

A. Ruszczyński and A. Świetanowski, "Accelerating the regularized decomposition method for two stage stochastic linear problems," *European Journal of Operational Research*, vol. 101, no. 2, pp. 328–342, 1997.

A. Régnier, "Détermination d'un horaire optimal d'arrêt des groupes turbo-alternateurs," Ph.D. dissertation, École Polytechnique de Montréal, 7 2008.

T. Santoso, S. Ahmed, M. Goetschalckx, and A. Shapiro, "A stochastic programming approach for supply chain network design under uncertainty," *European Journal of Operational Research*, vol. 167, no. 1, pp. 96–115, 2005.

J. L. Sawin, F. Sverrisson, K. Seyboth, R. Adib, H. E. Murdock, C. Lins, I. Edwards, M. Hullin, L. H. Nguyen, S. S. Prillianto *et al.*, "Renewables 2017 global status report," 2017.

S. Séguin, P. Côté, and C. Audet, "Self-scheduling short-term unit commitment and loading problem," *IEEE Transactions on Power Systems*, vol. 31, no. 1, pp. 133–142, 2016.

S. Séguin, S.-E. Fleten, P. Côté, A. Pichler, and C. Audet, "Stochastic short-term hydropower planning with inflow scenario trees," *European Journal of Operational Research*, vol. 259, no. 3, pp. 1156–1168, 2017.

A. Shapiro, D. Dentcheva, and A. Ruszczyński, *Lectures on stochastic programming: modeling and theory.* SIAM, 2009.

T. K. Siu, G. A. Nash, and Z. K. Shawwash, "A practical hydro, dynamic unit commitment and loading model," *IEEE Transactions on Power Systems*, vol. 16, no. 2, pp. 301–306, 2001.

S. Trukhanov, L. Ntaimo, and A. Schaefer, "Adaptive multicut aggregation for two-stage stochastic linear programs with recourse," *European Journal of Operational Research*, vol. 206, no. 2, pp. 395–406, 2010.

W. van Ackooij, R. Henrion, A. Möller, and R. Zorgati, "Joint chance constrained programming for hydro reservoir management," *Optimization and Engineering*, vol. 15, no. 2, pp. 509–531, 2014.

T. M. Welte, J. Vatn, and J. Heggset, "Markov state model for optimization of maintenance and renewal of hydro power components," in *Probabilistic Methods Applied to Power Systems, 2006. PMAPS 2006. International Conference on.* IEEE, 2006, pp. 1–7.

H. P. Williams, *Model building in mathematical programming.* John Wiley & Sons, 2013.

L. A. Wolsey, *Integer programming.* New York, NY, USA: Wiley-Interscience, 1998.

Z. A. Yamayee, "Maintenance scheduling: description, literature survey, and interface with overall operations scheduling," *IEEE Transactions on Power Apparatus and Systems*, no. 8, pp. 2770–2779, 1982.