UNIVERSITÉ DE MONTRÉAL

MÉTHODES DE DÉCOMPOSITION POUR LA PLANIFICATION À MOYEN TERME DE LA PRODUCTION HYDROÉLECTRIQUE SOUS INCERTITUDE

PIERRE-LUC CARPENTIER
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)
DÉCEMBRE 2013

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Cette thèse intitulée :

MÉTHODES DE DÉCOMPOSITION POUR LA PLANIFICATION À MOYEN TERME DE LA PRODUCTION HYDROÉLECTRIQUE SOUS INCERTITUDE

présentée par : CARPENTIER Pierre-Luc
en vue de l'obtention du diplôme de : Philosophiæ Doctor
a été dûment acceptée par le jury d'examen constitué de :

M. SOUMIS François, Ph.D., président
M. GENDREAU Michel, Ph.D., membre et directeur de recherche
M. BASTIN Fabian, Doctorat, membre et codirecteur de recherche
M. AUDET Charles, Ph.D., membre
M. BOUFFARD François, Ph.D., membre

DÉDICACE

À Anne-Marie, Philippe et Émilie...

REMERCIEMENTS

Je tiens tout d'abord à remercier mon directeur de recherche Michel Gendreau de m'avoir confié ce projet, pour le financement qu'il m'a accordé ainsi que pour le soutien qu'il m'a apporté durant l'ensemble du projet. Je remercie également mon co-directeur Fabian Bastin pour son implication dans ce projet de recherche. Je tiens aussi à remercier Grégory Émiel pour ses nombreuses suggestions et commentaires pertinents. Je remercie André Robitaille, Thérèse Falcon, Sylvain Robert, Laura Fagherazzi, Marko Blais, Éric Vatri, Nathalie Letendre, Patrick Jeandroz, Pierre Rivest, Albert Baho et François Shaiegetz d'Hydro-Québec Production pour leur grande disponibilité ainsi que pour leurs commentaires pertinents. Je remercie aussi le Fonds de recherche du Québec–Nature et technologies (FRQNT), le Conseil de recherches en sciences naturelles et en génie du Canada (CRSNG) et Hydro-Québec pour leur soutien financier. Je remercie aussi les étudiants, professeurs et employés de soutien du Centre interuniversitaire de recherche sur les réseaux d'entreprise, la logistique et le transport (CIRRELT) pour l'ambiance de travail durant la première année de mon doctorat. Je remercie aussi les membres du jury pour les commentaires et remarques qui ont permi d'améliorer la qualité de cette thèse.

RÉSUMÉ

Dans cette thèse, nous considérons le problème de planification à moyen terme (PPMT) de la production hydroélectrique sous incertitude visant la gestion de réservoirs sur un horizon de plusieurs mois. Nous nous intéressons particulièrement aux systèmes de haute dimension composés de dizaines de réservoirs et exploités par les grands producteurs hydroélectriques tels qu'Hydro-Québec. Ces producteurs exploitent des systèmes complexes à hauteur de chute et rendement variables avec des coûts fixes et des délais de démarrage et d'arrêt de groupes. En général, une grande variété de contraintes opérationnelles serrées doivent être satisfaites. Ces systèmes sont habituellement exploités dans un environnement décisionnel hautement incertain. Les modèles d'optimisation à moyen terme de la production considèrent généralement un pas de temps hebdomadaire ou mensuel et reposent sur une représentation simplifiée du système de production. Les courbes de production sont généralement convexes et linéaires par morceaux et les coûts fixe de transaction ou d'arrêt/démarrage sont généralement négligés. La principale source de complexité du PPMT et généralement attribuable à la prise en compte de l'incertitude. Les paramètres aléatoires du PPMT sont généralement caractérisés par une distribution de probabilité multidimensionnelle, continue et asymétrique. Ces caractéristiques sont difficiles à représenter de façon précise dans les modèles d'optimisation.

Au cours des dernières décennies, plusieurs méthodes d'optimisation stochastique ont été proposées dans la littérature en gestion de réservoirs. La plupart de ces méthodes sont limitées aux systèmes de dimension modeste en raison de la malédiction de la dimension. Les méthodes basées sur une représentation par arbre de scénarios de l'incertitude comptent parmi les rares approches qui sont applicables aux grands systèmes hydroélectriques. Ces méthodes fonctionnent en remplaçant la distribution continue d'origine par une distribution discrète contenant un nombre fini de réalisations possibles. Ainsi, le programme stochastique à résoudre peut être reformulé en un programme équivalent déterministe dont la taille est proportionnelle à la dimension du système contrôlé. La principale limitation de l'approche par arbre de scénarios est liée à l'augmentation exponentielle de la taille du programme équivalent déterministe avec le niveau de branchement de l'arbre. En pratique, le programme équivalent déterministe doit être résolu par une méthode de décomposition qui exploite sa structure mathématique spéciale. L'objectif de cette thèse consiste à développer et à évaluer différentes méthodes de décomposition permettant de résoudre le PPMT sous incertitude. La thèse est divisée en trois articles.

Le premier article démontre l'applicabilité de l'algorithme de *progressive hedging* (APH), une méthode de décomposition par scénario, pour faire la gestion de réservoirs hydroélectriques multiannuels dans un environnement hautement variable au Canada. L'APH est une méthode classique conçue pour résoudre des programme stochastique multiétape généraux posés sur un arbre de scénarios. Cette méthode fonctionne en appliquant une relaxation Lagrangienne augmentée aux contraintes de non-anticipativité (CNA) du programme stochastique. À chaque itération de l'APH, une série de sous-problèmes doivent être résolus. Chaque sous-problème correspond à une version déterministe du programme stochastique défini sur un scénario particulier de l'arbre. Des termes linéaires et quadratiques sont ajoutés à la fonction objectif des sous-problèmes afin de pénaliser les violations des CNA. Une des principales limitations de l'APH est liée à l'augmentation exponentielle du nombre de sous-problèmes et de termes de pénalité avec le niveau de branchement de l'arbre. Ce phénomène peut rendre l'application de l'APH particulièrement difficile lorsque l'arbre de scénarios considéré contient plusieurs étapes de branchement et couvre un horizon réparti sur des dizaines de périodes. Ces situations surviennent fréquemment lorsque des réservoirs multiannuels sont considérés. Une autre limitation importante de l'APH est causée par l'augmentation du niveau de difficulté des CNA avec la variabilité des scénarios contenus dans l'arbre. Ce phénomène complique l'application de l'APH dans les régions hydroclimatiques caractérisées par une forte variabilité saisonnière et interannuelle. Ces deux types de limitations peuvent ralentir considérablement le taux de convergence et le temps de calcul par itération de l'APH et rendre cette méthode inapplicable en pratique. Dans l'ensemble, très peu de chercheurs ont appliqué l'APH en gestion de réservoirs hydroélectriques. Les rares études portant sur ce type d'application considèrent un horizon de courte portée avec arbre de scénarios de petite taille avec un niveau de variabilité modeste. Dans cette étude, nous appliquons l'APH à la gestion de l'ensemble du parc d'Hydro-Québec sur un horizon de 92 semaines. L'arbre de scénarios considéré contient six étapes de branchement et 1635 noeuds. L'APH est particulièrement bien adaptée pour cette application étant donné du fait que la société d'État dispose actuellement d'un modèle déterministe opérationnel pour faire la planification à moyen terme de la production. En fait, seulement quelques modifications mineures sont nécessaires pour transformer le modèle déterministe actuel en un modèle stochastique basé sur l'APH.

Le deuxième article présente une nouvelle approche permettant de réduire le temps de calcul de l'APH lors la résolution de programme stochastique généraux. L'approche proposée fonctionne en appliquant un schéma de décomposition multiscénario au programme stochastique conçu de manière à minimiser le nombre de CNA auxquelles une RLA doit être appliquée. Chaque sous-problème prend la forme d'un programme stochastique défini

sur un groupe de scénarios. Les CNA liant les scénarios d'un même groupe sont représentées implicitement dans les sous-problèmes en adoptant une formulation par groupe de scénarios et par noeud plutôt qu'en utilisant le sytème d'indice traditionnel exprimé par période et par scénario. Seulement les CNA liant les différents groupes de scénarios sont représentées explicitement sous forme de contraintes d'égalité linéaires et relaxées. La méthode proposée est évaluée numériquement sur un problème de gestion de réservoirs hydroélectriques au Québec. Les résultats de cette expérience démontrent que notre méthode de partitionnement optimal a plusieurs avantages par rapport au schéma de décomposition par scénario traditionnel. Premièrement, elle permet de diminuer le temps de calcul par itération en réduisant le nombre de termes de pénalité à inclure dans les sous-problèmes et en réduisant le nombre de variables et contraintes dupliquées. Deuxièmement, notre approche permet d'accélérer le taux de convergence de l'algorithme en réduisant la variabilité des solutions intermédiaires obtenues aux noeuds dupliqués.

Le troisième article présente une extension de la méthode *L-Shaped* conçue spécifiquement pour faire la gestion de réservoirs hydroélectriques à haute capacité d'emmagasinnement. Lorsque de tels réservoirs sont considérés, l'horizon à moyen terme couvre typiquement plusieurs dizaines de périodes et les méthodes de décomposition conventionnelles telles que l'APH ne sont applicables que si un faible niveau de branchement est utilisé. Dans ces situations, l'arbre de scénarios considéré correspond généralement à une discrétisation très grossière de la distribution de probabilité continue sous-jacente. La méthode proposée dans cet article permet de considérer un niveau de branchement plus élevé que les méthodes conventionnelles le permettent. Pour atteindre cet objectif, nous posons l'hypothèse selon laquelle le processus stochastique décrivant les paramètres aléatoires subit une perte de mémoire à la période $t = \tau$. L'arbre de scénarios résultant de cette hypothèse possède une structure symétrique spéciale à la deuxième étape ($t > \tau$) que nous exploitons en appliquant un schéma de décomposition de Benders à deux étapes. Contrairement à la vaste majorité des méthodes de décomposition par étape proposées dans la littérature, chaque étape de décomposition de notre méthode correspond à plusieurs périodes consécutives. La méthode proposée fonctionne en construisant une fonction de recours convexe et linéaire par morceaux servant à représenter le coût espéré de deuxième étape (à $t > \tau$) en fonction de l'état du système à la fin le la première étape ($t = \tau$) dans le problème maître. Le sous-problème et le problème maître sont des programme stochastiqueM définis sur un sous-arbre et peuvent être résolus directement ou par une méthodes de décomposition conventionnelles. Nous démontrons l'efficacité de notre méthode en l'appliquant sur une version réduite du parc de production québécois sur un horizon de 104 semaines.

ABSRACT

In this thesis, we consider the midterm production planning problem (MTPP) of hydroelectricity generation under uncertainty. The aim of this problem is to manage a set of interconnected hydroelectric reservoirs over several months. We are particularly interested in high dimensional reservoir systems that are operated by large hydroelectricity producers such as Hydro-Québec. These producers operate a complex production system and must satisfy tight operational constraints in an highly uncertain decision environment. In general, midterm optimization models consider a weekly or monthly time step and rely on a simplified representation of the power system. The main source of complexity of the MTPP is usually related to the representation of uncertainty. Random parameters of the MTPP are usually characterized by a complex probability distribution function which is difficult to represent in numerical optimization models.

Over the past decades, several stochastic optimization methods were proposed in the literature for managing hydroelectric reservoirs over the midterm planning horizon. Most of these methods are only applicable on low- or medium-size systems due to the curse of dimensionality. stochastic optimization methods that are based on a scenario tree representation of uncertainty are among the rare approaches that can be used on large hydroelectric reservoir systems. These methods work by replacing the original continuous distribution by a discrete distribution possessing a finite number of possible realizations. The stochastic program to be solved can be reformulated into a deterministic equivalent program whose size is proportional to the system size. The main limitation with this approach is due to the exponential growth of the DEP's size with the branching level of the tree. In practice, the DEP is usually quite large and must be solved using a decomposition method which exploits its special mathematical structure. The aim of this thesis is to develop and evaluate different decomposition methods for solving the MTPP under uncertainty. This thesis is divided in three articles.

The first article demonstrates the applicability of the progressive hedging algorithm (PHA), a scenario decomposition method, for managing hydroelectric reservoirs with multiannual storage capacity under highly variable operating conditions in Canada. The PHA is a classical stochastic optimization method designed to solve general multistage stochastic programs defined on a scenario tree. This method works by applying an augmented Lagrangian relaxation on non-anticipativity constraints (NACs) of the stochastic program. At each iteration of the PHA, a sequence of subproblems must be solved. Each subproblem

corresponds to a deterministic version of the original stochastic program for a particular scenario in the scenario tree. Linear and a quadratic terms must be included in subproblem's objective functions to penalize any violation of NACs. An important limitation of the PHA is due to the fact that the number of subproblems to be solved and the number of penalty terms increase exponentially with the branching level in the tree. This phenomenon can make the application of the PHA particularly difficult when the scenario tree covers several tens of time periods. Another important limitation of the PHA is caused by the fact that the difficulty level of NACs generally increases as the variability of scenarios increases. Consequently, applying the PHA becomes particularly challenging in hydroclimatic regions that are characterized by a high level of seasonal and interannual variability. These two types of limitations can slow down the algorithm's convergence rate and increase the running time per iteration. Overall, very few researchers applied the PHA on reservoir management problems. The few studies that consider this type of application consider a short-range horizon with a small scenario tree. In this study, we apply the PHA on Hydro-Québec's power system over a 92-week planning horizon. Hydrologic uncertainty is represented by a scenario tree containing 6 branching stages and 1,635 nodes. The PHA is especially well-suited for this particular application given that the company already possess a deterministic optimization model to solve the MTPP. In fact, only a few minor modifications are required to transform the current model into a new PHA-based stochastic optimization.

The second article presents a new approach which enhances the performance of the PHA for solving general Mstochastic programs. The proposed method works by applying a multiscenario decomposition scheme on the stochastic program. Our heuristic method aims at constructing an optimal partition of the scenario set by minimizing the number of NACs on which an augmented Lagrangean relaxation must be applied. Each subproblem is a stochastic program defined on a group of scenarios. NACs linking scenarios sharing a common group are represented implicitly in subproblems by using a group-node system index instead of the traditional scenario-time index system. Only the NACs that link the different scenario groups are represented explicitly and relaxed. The proposed method is evaluated numerically on an hydroelectric reservoir management problem in Québec. The results of this experiment show that our method has several advantages. Firstly, it allows to reduce the running time per iteration of the PHA by reducing the number of penalty terms that are included in the objective function and by reducing the amount of duplicated constraints and variables. In turn, this allows to reduce the running time per iteration of the algorithm. Secondly, it allows to increase the algorithm's convergence rate by reducing the variability of intermediary solutions at duplicated tree nodes. Thirdly, our approach reduces the amount of random-access

memory (RAM) required for storing Lagrange multipliers associated with relaxed NACs.

The third article presents an extension of the L-Shaped method designed specifically for managing hydroelectric reservoir systems with a high storage capacity. When such systems are considered, the midterm planning horizon usually contains several tens of time periods and conventional decomposition methods such as the PHA can only be used if a low branching level is used. In these situations, the scenario tree typically corresponds to a very coarse representation of the underlying continuous probability distribution. The method proposed in this paper enables to consider a higher branching level than conventional decomposition method enables. To achieve this, we assume that the stochastic process driving random parameters has a memory loss at time period $t = \tau$. Because of this assumption, the scenario tree possess a special symmetrical structure at the second stage ($t > \tau$). We exploit this feature using a two-stage Benders decomposition method. Contrary to most stage-wise decomposition methods that were proposed in previous studies, each decomposition stage covers several consecutive time periods. The proposed method works by constructing a convex and piecewise linear recourse function that represents the expected cost at the second stage in the master problem. The subproblem and the master problem are stochastic program defined on scenario subtrees and can be solved using a conventional decomposition method or directly. We test the proposed method on an hydroelectric power system in Québec over a 104-week planning horizon.

TABLE DES MATIÈRES

LISTE DES TABLEAUX

LISTE DES FIGURES

LISTE DES SIGLES ET ABRÉVIATIONS

ADP	Approximate DP
APH	Algorithme de progressive hedging
CIRRELT	Centre interuniversitaire de recherche sur les reseaux d'entreprise, la logistique et le transport
CNA	Contraintes de non-anticipativité
CNAR	CNA relaxée
CRSNG	Conseil de recherches en sciences naturelles et en génie du Canada
DEP	Deterministic equivalent program
DOM	Deterministic OM
DP	Dynamic Programming
DSDP	Discrete SDP
ELS	Extended L-Shaped
ES	Expected sales
ETG	Expected thermal generation
EVPI	Expected value of perfect information
FRQNT	Fonds de recherche du Québec–Nature et technologies
HDRS	High dimensional reservoir system
HQRMP	Hydro-Québec's RMP
MLA	Memory loss approximation
MPAR(p)	Multivariate period AR model of order p
MSP	Multistage SP
MTPP	Midterm production planning problem
MWV	Marginal water value
NAC	Non-anticipativity constraints
NBD	Nested Benders Decomposition
NDP	Neuro DP
PC	Personnal computer
PHA	Progressive hedging algorithm
PPD	Planification de la production en contexte déterministe
PPMT	Problème de planification à moyen terme
RF	Reduction factor
RAM	Random-access memory
RMP	Reservoir management problem

SAMS	Stochastic Analysis Modeling and Simulation
SDDP	Stochastic Dual DP
SDP	Stochastic DP
SOM	Stochastic OM
SP	Stochastic program
ST	Scenario tree
STBM	ST-based method
VSS	Value of stochastic solution

CHAPITRE 1

INTRODUCTION

L'industrie de l'énergie est, dans son ensemble, un secteur très riche en applications pour les différentes méthodes d'optimisation stochastique (Wallace et Fleten, 2003; Kallrath *et al.*, 2009; Sagastizabal, 2012). Au cours des vingt dernières années, ces méthodes ont été particulièrement populaires dans le domaine de la production hydroélectrique. Les grands producteurs d'hydroélectricité au Canada (Fortin, 2008) et ailleurs dans le monde font face à une multitude de problèmes d'optimisation sous incertitude liés au design, à la gestion de la production et à la gestion de l'entretien de systèmes énergétiques. Ces producteurs exploitent des systèmes complexes composés de plusieurs dizaines de composantes contrôlables et doivent composer avec plusieurs sources d'incertitude. Parmi les paramètres incertains typiquement rencontrés en pratique, on compte notamment la demande électrique (puissance) à satisfaire, les apports hydriques naturels aux réservoirs, le niveau de production intermittente, le prix de l'énergie sur les marchés déréglementés et les défaillances de composantes. En général, une variété de contraintes opérationnelles de nature énergétique, hydrique et financière doivent être satisfaites. Dans bien des cas, la production hydroélectrique doit être coordonnée avec des moyens de production complémentaires et avec les achats et ventes d'énergie sur différents marchés. La grande flexibilité de la production hydroélectrique est souvent utilisée pour offrir une variété de services auxiliaires sur les marchés dérèglementés de l'électricité (Rebours *et al.*, 2007a,b).

1.1 Problématique

Les problèmes de gestion optimale de la production hydroélectrique sont typiquement classés selon l'horizon temporel considéré (Wallace et Fleten, 2003). Cette thèse porte sur le problème de planification à moyen terme (PPMT) sous incertitude. L'objectif principal du PPMT consiste essentiellement à gérer le volume d'eau entreposé dans un ensemble de réservoirs interconnectés. L'horizon de planification à moyen terme couvre typiquement plusieurs mois voire quelques années. Les deux principales sources de complexité du PPMT sont attribuables à la prise en compte de l'incertitude et à la dimension du système exploité. Les paramètres aléatoires du PPMT évoluent habituellement selon un processus stochastique multidimensionnel et sont caractérisés par une distribution de probabilité (jointe) continue.

La plupart des modèles d'optimisation à moyen terme (MOMT) de la production hydroélectrique utilisent un pas de temps hebdomadaire ou mensuel et reposent sur une représentation simplifiée du système de production. En pratique, les gestionnaires de production utilisent les MOMT pour effectuer différentes analyses énergétiques, hydriques ou économiques. Sur le plan opérationnel, les MOMT servent aussi à encadrer l'ordonnancement de la production à court terme. En fait, ces modèles permettent de calculer la valeur marginale de l'eau (Tilmant *et al.*, 2008) ou une cible de production hebdomadaire (soutirage, volume de ventes) qui est fournie à un modèle hautement détaillé d'optimisation à court terme de la production (Shawwash *et al.*, 2000; Fleten et Kristoffersen, 2008).

1.2 Modélisation

1.2.1 Description générale

L'objectif du PPMT consiste à gérer un parc de production à prédominance hydroélectrique sur un horizon fini de T périodes. Le parc de production considéré contient des centrales hydroélectriques $i = 1, ..., I$ alimentées par un ensemble de réservoirs interconnectés $j = 1, ..., J$. Le système de production peut aussi contenir d'autres composantes contrôlables telles que des évacuateurs de crue, des centrales thermiques et des marchés d'achat et de vente d'électricité. Les grands systèmes contiennent typiquement plusieurs dizaines voire quelques centaines de composantes contrôlables. Un ensemble de contraintes opérationnelles doivent être satisfaites. Ces contraintes peuvent être de nature énergétique, hydrique ou financière. Dans bien des cas, une demande électrique (puissance) doit être satisfaite et la production hydroélectrique doit être coordonnée avec d'autres moyens de production contrôlables, prédéterminés ou intermittents. Pour les grands systèmes, la puissance produite par chaque centrale hydroélectrique i est généralement modélisée par une fonction concave et linéaire par morceaux qui dépend du débit turbiné, du volume d'eau contenu dans les réservoirs amont et aval et, dans certains cas, du débit déversé (Diniz et Maceira, 2008).

1.2.2 Processus stochastique

Un vecteur aléatoire ξ_t est défini à chaque période $t = 1, ..., T$ de l'horizon. Chaque composante de ξ_t correspond à un paramètre aléatoire du problème tels que les apports naturels dans un réservoir, la demande électrique ou le prix de l'énergie à un noeud du réseau. Les composantes peuvent être corrélées entre elles à une période donnée. Par exemple, l'intensité des apports naturels d'un réservoir n'est généralement pas indépendante des apports observés dans un autre réservoir. De plus, les vecteurs aléatoires évoluent selon un processus

stochastique multidimensionnel

$$\{\xi_t : t = 1, ..., T\}.$$

On suppose que la distribution de probabilité $\mathbb{P}_t(\ \cdot \mid \xi_{t-1}, ..., \xi_{t-p})$ décrivant ξ_t est connue et dépend de l'historique observé aux p dernières périodes. En pratique, des effets de persistence sont généralement observés sur les apports, les prix et la demande électrique et l'historique observé donne une information sur la tendance suivie par le processus. On suppose aussi que la distribution $\mathbb{P}_t$ n'est pas influencée par les décisions prises par les gestionnaires.

1.2.3 Formulation stochastique

Le PPMT prend la forme du programme stochastique suivant :

$$(\mathcal{P}) \qquad \min_{x_t \in \mathbb{R}_n} \mathbb{E}\left[\sum_{t=1}^{T} g_t(x_t, \xi_t)\right] \tag{1.1}$$

sous les contraintes

$$A_t(\xi_t)x_t + B_t(\xi_t)x_{t-1} = b_t(\xi_t) \quad , \quad \forall t = 1, ..., T \tag{1.2}$$

$$\underline{x}_t \leq x_t \leq \overline{x}_t \quad , \quad \forall t = 1, ..., T \tag{1.3}$$

$$x_t \in \mathcal{F}_t(\xi_1, ..., \xi_t) \quad , \quad \forall t = 1, ..., T. \tag{1.4}$$

La fonction objectif (1.1) à minimiser est le coût total espéré. L'espérance mathématique $\mathbb{E}[\cdot]$ est calculée par rapport à la distribution de probabilité des vecteurs aléatoires ξ_t où les fonctions de coûts $g_t(x_t, \xi_t)$ aux périodes $t = 1, ..., T$ sont convexes et linéaires par morceaux. Les composantes du vecteur x_t correspondent aux variable de décision du problème à la période t. Les contraintes (1.2) servent typiquement à représenter différentes contraintes physiques. A_t, B_t sont des matrices et b_t est un vecteur dont les coefficients sont aléatoires. Les bornes (1.3) servent à représenter les limites physiques du système ou des contraintes de nature opérationnelle. Les contraintes de non-anticipativité (1.4) (CNA) garantissent que chaque décision x_t ne peut dépendre que de l'information $\xi_1, ..., \xi_t$ qui est disponible à la période t. Chaque ensemble $\mathcal{F}_t(\xi_1, ..., \xi_t)$ contient toutes les commandes non-anticipatives à la période t. Pour être non-anticipative, une solution ne doit pas exploiter des informations qui sont inconnues du décideur au moment de choisir x_t. Le problème $\mathcal{P}$ prend la forme d'un programme stochastique multiétape de haute dimension. La principale source de complexité de $\mathcal{P}$ est due au fait que la distribution de probabilité jointe décrivant les vecteurs aléatoires ξ_t est continue et très complexe (corrélations, asymétrie, multidimensionnelle...). Conséquemment, la décision $x_t(s_t)$ est une fonction définie sur un état continu du système $s_t = (v_t, \xi_{t-1}, ..., \xi_{t-p})$ où v_t est

un vecteur représentant le volume d'eau dans chaque réservoir. L'espérance correspond à une intégrale multiple définie sur un domaine de grande dimension, ce qui complique le problème $\mathcal{P}$.

1.3 Objectifs de recherche

L'objectif général de cette thèse consiste à développer et à évaluer différentes méthodes d'optimisation stochastique permettant de résoudre efficacement le problème $\mathcal{P}$. Nous nous limitons aux méthodes qui sont applicables aux systèmes de haute dimension exploités par les grands producteurs hydroélectriques tels qu'Hydro-Québec. Pour de tels systèmes, chaque vecteur x_t peut contenir des centaines de variables de décision à la période t. Les problèmes auxquels font face ces producteurs sont réputés comme étant particulièrement difficile à résoudre en raison du niveau d'incertitude élevé caractérisant l'environnement décisionnel ainsi qu'en raison de la grande complexité (rendement et hauteur de chute variable), de la dimension élevée et de la grande capacité d'emmagasinement du système hydroélectrique contrôlé. Ces systèmes contiennent généralement des réservoirs possédant un cycle de remplissage saisonnier ou multiannuel et sont opérés sous des conditions opérationnelles hautement variables d'une saison/année à l'autre. Dans bien des cas, l'horizon de planification à moyen terme doit couvrir plusieurs mois, voire quelques années.

1.4 Structure de la thèse

La thèse est organisée de la façon suivante. Le chapitre 2 présente une revue de littérature sur les méthodes d'optimisation sous incertitude pour la gestion de réservoirs hydroélectriques. Nous présentons la démarche utilisée dans l'ensemble de la thèse au chapitre 3. Les chapitres 4, 5 et 6 présentent trois articles scientifiques ayant été rédigés dans le cadre de cette thèse. Une discussion et une conclusion sont présentées au chapitre 8.

CHAPITRE 2

REVUE DE LA LITTÉRATURE

Depuis le début des années 1960, plusieurs types d'approches ont été proposés dans la littérature pour effectuer la gestion optimale de réservoirs hydroélectriques sur l'horizon à moyen terme. Les articles de Yeh (1985), Labadie (2004) et Rani et Moreira (2010) présentent une revue détaillée des principales études passées dans ce domaine. Malgré l'étendue de la littérature, un grand fossé existe toujours entre la recherche académique et les applications industrielles comme l'indique Labadie (2004). Ce phénomène s'explique en partie par la grande complexité mathématique caractérisant plusieurs méthodes proposées, ce qui les rend difficiles à comprendre et interpréter. De plus, les méthodes proposées dans la littérature sont souvent évaluées sur des études de cas hautement simplifiées par rapport aux problèmes industriels. Aussi, le développement de modèles opérationnels nécessite généralement un investissement important de ressources humaines, monétaires et informatiques dont ne disposent pas les équipes chargées du développement d'outils d'aide à la décision.

Wurbs (1993) souligne le fait que le choix de la méthode idéale pour résoudre un problème particulier dépend des préférences des gestionnaires et des caractéristiques de l'application considérée. L'applicabilité de chaque méthode dépend notamment

- de la dimension du système considéré ;
- du niveau de complexité du système considéré ;
- du niveau de détail requis pour représenter adéquatement les paramètres incertains (nombre de paramètres aléatoires, autocorrélation spatio-temporelle, discrétisation de la distribution) ;
- et du type de solution recherchée par les gestionnaires.

2.1 Prise en compte de l'incertitude

Trois types d'approches différentes sont communément utilisées pour prendre en compte l'incertitude en gestion optimale de réservoirs. La premier type d'approche est la plus rudimentaire des trois et consiste simplement à lancer manuellement un modèle d'optimisation déterministe de façon répétée avec un ou quelques réalisations possibles du processus stochastique $\{\xi_t : t = 1, ..., T\}$. L'avantage principal de cette approche consiste à permettre aux gestionnaires d'obtenir une solution optimale pour chaque scénario très rapidement avec des

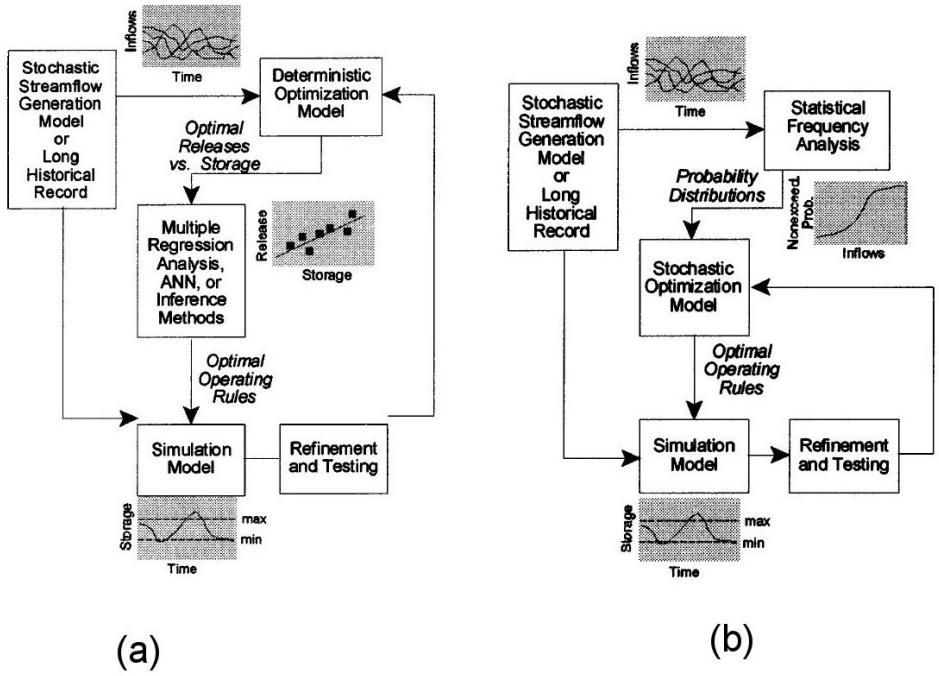

Figure 2.1 Méthode d'optimisation stochastique implicite (a) et explicite (b).

ressources informatiques modestes. À l'heure actuelle, les solveurs commerciaux permettent de résoudre directement les programmes mathématiques généraux de grande taille en très peu de temps. En général, la taille du programme déterministe à résoudre est proportionnelle à la dimension du système contrôlé. Conséquemment, l'approche déterministe est généralement applicable aux systèmes de haute dimension. Malheureusement, en négligeant complètement l'incertitude, les modèles d'optimisation déterministes retournent généralement des solutions sous-optimales comme l'indiquent Philbrick et Kitanidis (1999). De plus, les solutions déterministes varient considérablement d'un scénario à l'autre et ceci complique significativement la tâches des gestionnaires pour l'interprétation des différents résultats obtenus. En pratique, des contraintes additionnelles doivent souvent être ajoutées au modèle d'optimisation déterministe par les gestionnaires afin d'éliminer les solutions trop optimistes du domaine réalisable. Malheureusement, la calibration de telles contraintes est une tâche délicate et exigeante pour les gestionnaires de systèmes hydriques. D'un côté, l'utilisation de contraintes trop conservatrices peut augmenter inutilement les coûts de gestion (inondations, production thermique). À l'inverse, l'utilisation de contraintes trop agressives expose les gestionnaires à un niveau de risque trop élevé.

Le deuxième type d'approche pour prendre en compte l'incertitude consiste à utiliser une

méthode d'optimisation stochastique implicite. Ce type de méthode correspond à une version plus sophistiquée du premier type d'approche et fonctionne essentiellement en lançant un modèle déterministe d'une façon automatisée pour un grand nombre de scénarios différents. Un vecteur d'état s_t et un vecteur de contrôle sont définis à chaque période t. Chaque composante du vecteur de contrôle représente une variable de décision prise à la période t. Chaque composante du vecteur d'état est une source d'information disponible au décideur à la période t. Les solutions déterministes obtenues sont utilisées pour calibrer, par régression, une politique de gestion $\pi = (\mu_1(s_1), ..., \mu(s_T))$ composée de règles de décision $\mu_t : S_t \to U_t$ retournant le vecteur de contrôle $u_t \in U_t(s_t)$ à appliquer pour n'importe quel vecteur d'état possible du système $s_t \in S_t$ au début de chaque période $t = 1, ..., T$. Le vecteur de contrôle u_t contient toutes les décisions nécessaires pour opérer le système complet à la période t. Le vecteur d'état s_t doit contenir toutes les informations disponibles au gestionnaire au moment où u_t est choisi. Le vecteur d'état contient typiquement le volume d'eau entreposé dans chaque réservoir et l'historique $\xi_t, ..., \xi_{t-p}$ du processus stochastique observé aux p dernières périodes. La Figure 2.1a illustre le fonctionnement des méthodes implicites.

En principe, n'importe quel modèle d'optimisation déterministe peut être utilisé pour calibrer la politique π. Bhaskar et Whitlatch (1980); Karamouz et Houck (1982) ont utilisé une méthode implicite pour faire la gestion de systèmes de faible dimension. En principe, des systèmes de haute dimension peuvent aussi être traités par une méthode implicite. Par contre, l'application de telles méthodes devient particulièrement difficile lorsque des contraintes opérationnelles doivent être satisfaites. De plus, il n'existe aucune garantie sur la qualité de la règle de décision calibrée. Dans certaines situations, la qualité de toutes les solutions peut être très mauvaise en raison de l'hypothèse déterministe. Aussi, la régression peut être mauvaise lorsque les solutions déterministes sont drastiquement différentes.

Le troisième type d'approche est plus sophistiqué que les deux premiers et consiste à utiliser une méthode d'optimisation stochastique explicite. La formulation mathématique des méthodes explicites repose essentiellement sur une description probabiliste des paramètres aléatoires. La Figure 2.1b illustre le fonctionnement des méthodes explicites. La prise en compte explicite de l'incertitude permet généralement d'obtenir des solutions de qualité supérieure à celles retournées par les modèles déterministes en gestion de réservoirs (Philbrick et Kitanidis, 1999). De plus, les solutions stochastiques sont généralement plus faciles à interpréter que les solutions déterministes par les gestionnaires de production. L'approche explicite réduit considérablement le besoin d'ajouter des contraintes additionnelles au modèle d'optimisation, ce qui simplifie la tâche des gestionnaires de production. En pratique, la calibration

de telles contraintes est une tâche délicate et exigeante pour les gestionnaires de systèmes hydriques. Les MOS sont devenues particulièrement populaires en gestion de réservoirs à partir des années 1980. Dans l'ensemble, deux familles de méthodes explicites ont été proposées dans la littérature : les méthodes basées sur la programmation dynamique et les méthodes basées sur un représentation par arbre de scénarios de l'incertitude. Les deux sections qui suivent donnent une vue d'ensemble de ces deux familles de méthodes explicites.

2.2 Approches par programmation dynamique

2.2.1 Cadre théorique

Pour utiliser une approche par programmation dynamique, on doit ramener le problème $\mathcal{P}$ sous la forme d'un processus de décision markovien. Chaque vecteur de décision $x_t = (\psi_t, u_t)$ est partitionné en deux vecteurs ψ_t et u_t. Le vecteur ψ_t représente l'état physique du système contrôlé (p. ex. volume d'eau dans chaque réservoir, débit en transit sur cours d'eau...) au début de la période t tandis que u_t est le vecteur de contrôle du système. Chaque composante de u_t correspond à une variable contrôlé à la période t (p. ex. débit turbiné/déversé, puissance produite...). À chaque période $t = 1, ..., T$, un coût $g_t(\psi_t, u_t, \xi_t)$ est engendré et l'état physique évolue selon la fonction de transition

$$\psi_{t+1} = f_t(\psi_t, u_t, \xi_t).$$

Ensuite, on définit un vecteur d'état $s_t \in S_t \subset \mathbb{R}^m$ contenant toute l'information connue à la période t qui influence le coût espéré futur aux périodes $t, t+1, ..., T$. Le vecteur d'état $s_t = (\psi_t, \xi^t)$ contient typiquement l'état physique courant ψ_t et un vecteur ξ^t contenant des informations (observations ou prévision) sur la tendance de $\xi_t, ..., \xi_T$.

Les méthodes de programmation dynamique visent essentiellement à construire une politique $\pi = (\mu_1(s_1), ..., \mu(s_T))$ retournant la commande $u_t = \mu_t(s_t) \in U_t(s_t)$ à appliquer pour n'importe quel état possible du système $s_t \in S_t$ à chaque période $t = 1, ..., T$.

Les méthodes de programmation dynamique fonctionnent en construisant une fonction de Bellman $\mathcal{Q}_t(s_t)$ à chaque période $t = 1, ..., T$ de l'horizon de planification. Chaque fonction $\mathcal{Q}_t(s_t)$ retourne le coût espéré optimal aux périodes $t, t+1, ..., T$ conditionnel à l'état courant du système s_t. Ces fonctions sont construites en résolvant les équations de Bellman

$$\mathcal{Q}_t(s_t) = \min_{u_t \in U_t(s_t)} \mathbb{E}\left[g_t(\psi_t, u_t, \xi_t) + \mathcal{Q}_{t+1}(f_t(\psi_t, u_t, \xi_t))\right], \qquad \forall s_t \in S_t$$

par chaînage arrière pour $T, T-1, ..., 1$.

2.2.2 Méthodes de résolution

Une grande variété de méthodes d'optimisation basées sur le principe de la programmation dynamique (Bellman, 1957) a été proposée pour faire la gestion de réservoirs (Yakowitz, 1982; Nandalal et Bogardi, 2007). L'algorithme de programmation dynamique stochastique discrète classique nécessite que les espaces d'état S_t et de contrôle U_t soient discrétisés. Cette méthode de résolution retourne une solution exacte pour le problème discrétisé et ne requiert aucune hypothèse simplificatrice sur les propriétés mathématiques des fonctions de coûts g_t et des contraintes. De plus, la complexité algorithmique de la PDSD augmente linéairement avec le nombre de périodes T. Conséquemment, la programmation dynamique stochastique discrète est très bien adaptée pour la gestion de systèmes hautement complexes lorsque l'horizon temporel couvre plusieurs dizaines de périodes. Malheureusement, la complexité algorithmique de la programmation dynamique stochastique discrète augmente exponentiellement avec la dimension des vecteurs d'état et de contrôle *malédiction de la dimension*. Pour illustrer ce phénomène, supposons que $s_t \in \mathbb{R}^m$ et que chaque variable d'état peut prendre M valeurs différentes. Alors, le nombre total de sous-problèmes à résoudre est TM^m. Supposons aussi que $u_t \in \mathbb{R}^r$ et que chaque variable de contrôle peut prendre R valeurs différentes. Alors, le nombre d'opérations arithmétiques par sous-problème est proportionnel à R^r et le nombre total d'opérations de l'algorithme de PDSD sera proportionnel à TM^mR^r. Si $T = M = m = R = r = 10$, les nombre d'opérations sera proportionnel à 10^{21}. Sur un processeur de 3 GHz, le temps de calcul sera proportionnel à 10^4 années de calcul. En pratique, la PDSD n'est donc applicable que sur des systèmes de petite dimension (4 variables d'état ou moins). La programmation dynamique stochastique discrète a été appliquée en gestion de réservoirs à plusieurs reprises (p. ex. Tejada-Guibert *et al.*, 1995).

Différentes méthodes de programmation dynamique approchée ont été proposées afin de considérer des systèmes de dimension plus élevée. Le livre de Powell (2011) présente une introduction à ce type de méthode. Une des approches possibles permettant de réduire la dimension de s_t et u_t consiste à agréger différents réservoirs et à appliquer la PDS classique sur le système transformé. Ce type d'approche a été utilisé en gestion de réservoirs par Archibald *et al.* (1997) et Turgeon (1998). Les méthodes de programmation dynamique neuronales (Bertsekas et Tsitsiklis, 1996) constituent une approche prometteuse permettant de traiter des systèmes de dimension supérieure. Castelletti *et al.* (2007) a utilisé un algorithme de programmation dynamique neuronales en gestion de réservoirs. Lee et Labadie (2007) et Castelletti *et al.* (2010) ont appliqué une méthode d'apprentissage par renforcement en ges-

tion de réservoirs.

L'algorithme de programmation dynamique duale stochatique proposée par Pereira et Pinto (1991) est un autre type de méthode de programmation dynamique approchée qui a été largement utilisée pour la gestion de grands systèmes. Cette méthode évite la malédiction de la dimension associée aux espaces d'état et de contrôle en traitant ces ensembles de façon continue. À chaque itération de la programmation dynamique duale stochatique, deux phases sont effectuées. La première phase consiste à construire une approximation externe convexe et linéaire par morceaux de $\mathcal{Q}_t$ en résolvant les équations de Bellman par chaînage arrière pour un nombre restreint d'états $s_t \in S_t$. La deuxième phase consiste à générer de nouvelles trajectoires S_t en utilisant l'approximation courante de $\mathcal{Q}_t$ dans les équations de Bellman pour $t = 1, ..., T$ (chaînage avant). Le processus stochastique $\{\xi_t : t = 1, ..., T\}$ est représenté par un processus stochastique linéaire

$$\xi_t = \sum_{i=1}^{p} \Phi_{it} \xi_{t-i} + \epsilon_t$$

où ϵ_t représente un bruit dont la distribution est connue et Φ_{it} sont des matrices aux coefficients connus. La programmation dynamique duale stochatique a d'abord été développée au Brésil (Maceira *et al.*, 2008). Par la suite, cette méthode a été appliquée sur le système norvégien (e.g. Rotting et Gjelsvik, 1992; Gjelsvik *et al.*, 2010) et sur d'autres systèmes ailleurs dans le monde (p. ex. Tilmant et Kelman, 2007; Tilmant *et al.*, 2008; Goor *et al.*, 2011).

2.3 Approches par arbre de scénarios

2.3.1 Modélisation de l'incertitude

Pour utiliser ce type de représentation de l'incertitude, on suppose essentiellement que chaque vecteur aléatoire ξ_t possède un nombre fini de réalisations possibles. On suppose aussi que la distribution de probabilité $\mathbb{P}_t$ est indépendante des variables de décision. Ces hypothèses permettent de représenter le processus stochastique $\{\xi_t : t = 1, ..., T\}$ au moyen d'un arbre de scénarios $\mathcal{T}$ possédant un nombre fini de noeuds $n \in \mathcal{N}$. Chaque scénario $\omega \in \Omega$ de l'arbre représente une réalisation possible du processus stochastique et correspond à un chemin partant de la racine $0 \in \mathcal{N}$ et allant jusqu'à une feuille $\ell(\omega)$. Un vecteur ξ_n et une probabilité p_n sont définis à chaque noeud n. La topologie de l'arbre est caractérisée par une fonction $a(n)$ retournant l'ancêtre du noeud n.

Différentes méthodes ont été proposées pour construire un arbre de scénarios à partir de

scénario historiques ou synthétiques (Høyland et Wallace, 2001; Pflug, 2001; Dupačová *et al.*, 2000; Gröwe-Kuska *et al.*, 2003; Heitsch et Römisch, 2009). Heitsch et Römisch (2009) ont proposé une méthode de construction efficace basée sur les résultats théoriques obtenus par Heitsch *et al.* (2006). Le logiciel SCENRED2 faisant partir du General Algebraic Modeling System (GAMS) est une implémentation informatique de cette technique.

2.3.2 Programme équivalent déterministe

La représentation par arbre de scénarios de l'incertitude permet transformer le programme stochastique $\mathcal{P}$ en un programme équivalent déterministe

$$(\mathcal{E}) \qquad \min_{x_n \in \mathbb{R}_n} \sum_{n \in \mathcal{N}} p_n g_{t(n)}(x_n, \xi_n)$$

sous les contraintes

$$A_n x_n + B_n x_{a(n)} = b_n \quad , \quad \forall n \in \mathcal{N}, \tag{2.1}$$

$$\underline{x}_{t(n)} \leq x_n \leq \overline{x}_{t(n)} \quad , \quad \forall n \in \mathcal{N}. \tag{2.2}$$

Le programme $\mathcal{E}$ est obtenu en appliquant les transformations suivantes à $\mathcal{P}$. L'opérateur d'espérance utilisé dans (1.1) est remplacé par une somme finie de termes dans (2.3.2). Les vecteurs de décision x_t aux périodes $t = 1, ..., T$ sont remplacés par un nouveau vecteur de décision x_n aux noeuds $n \in \mathcal{N}$. La fonction $t(n)$ retourne la période associée au noeud n. Les contraintes physiques (2.1) et (2.2) au noeud n correspondent aux contraintes (1.2) et (1.3) à la période $t(n)$, respectivement. La solution du programme $\mathcal{E}$ prend la forme d'un arbre de décision possédant une structure de branchement identique à celle de l'arbre de scénarios utilisé pour représenter la variabilité des paramètres aléatoires. Chaque vecteur de décision x_n correspond à la décision à prendre lorsque le processus stochatique pase par le noeud noeud n. L'indice du noeud n ne contient que l'historique observé du processus stochastique aux périodes $1, 2, ..., t(n)$ et ne dépend pas des réalisations futures à $t+1, t+2, ..., T$. Conséquemment, n'importe quelle solution réalisable x_n n'exploite que des informations qui sont disponible au décideur à la période $t(n)$. Les commandes x_n contenues dans l'arbre de décision satisfont donc les contraintes de non-anticipativité (1.4).

En général, le programme $\mathcal{E}$ est de grande taille et ne peut pas être résolu directement. Différentes méthodes de décomposition ont été proposées au cours des dernières décennies pour résoudre efficacement ce type de problème. Ces méthodes fonctionnent en exploitant la

structure mathématique particulière de nombreux problèmes rencontrés en pratique. Le livre de Conejo *et al.* (2006) présente une introduction aux différentes méthodes de décomposition appliquées à des problèmes généraux. Certaines des méthodes générales ont été spécialisées pour la résolution de programmes stochastiques (Rusczynzki, 2003). L'article de Sagastizabal (2012) donne une vue d'ensemble des principales méthodes de décomposition utilisées dans le domaine de l'énergie. Deux strarégies de décomposition sont communément utilisées pour la résolution de programmes stochastique. La première stratégie est la plus répandue et consiste à appliquer une décomposition de Benders (1962). La deuxième stratégie consiste à appliquer une décomposition par scénarios. Les deux sous-sections qui suivent présentent chacune de ces approches.

2.3.3 Décomposition de Benders

Plusieurs différentes méthodes de programmation stochastique à deux et plusieurs étapes basées sur la décomposition de Benders (1962) ont été proposées au cours des dernières décennies. Ces méthodes appliquent un schéma de décomposition par étape au programme $\mathcal{E}$ et fonctionnent en construisant une série de fonctions de recours $\mathcal{Q}_n$ convexes et linéaires par morceaux définie aux noeuds $n \in \mathcal{N}$ de l'arbre. Cette fonction est représentée par un nombre fini de coupes de Benders qui sont représentées au moyen de contraintes d'inégalité linéaires. Les coefficients de chaque coupe sont obtenus en exploitant la solution duale des sous-problèmes. Dans la plupart des méthodes, chaque étape de décomposition correspond à une étape de branchement dans l'arbre. La méthode *L-Shaped* classique proposée par Van Slyke et Wets (1969) permet de résoudre des PS linéaires définis sur un arbre de scénarios à deux étapes en exploitant la structure en forme de L des éléments non nuls dans la matrice de contraintes. Birge et Louveaux (1988) ont proposé une version multicoupe de cet algorithme. Une version en nombres entiers de la méthode *L-Shaped* a été proposée par Laporte et Louveaux (1993). Birge (1985) a proposé une extension de la méthode *L-Shaped* permettant de résoudre les programmes définis sur un arbre de scénarios contenant plusieurs étapes. Cette méthode a été appliquée à des problèmes de gestion de réservoirs à différentes reprises (Jacobs *et al.*, 1995; dos Santos et Diniz, 2009; Archibald *et al.*, 1999). Küchler et Vigerske (2007) a proposé une extension la décomposition de Benders multiétape pour traiter des arbre de scénarios recombinants. M. L. L. dos Santos et Goncalves (2009) ont proposé une version de la méthode de Benders où chaque étape de décomposition correspond à un bloc de périodes.

2.3.4 Décomposition par scénario

L'algorithme de *progressive hedging* (APH) proposé par Rockafellar et Wets (1991) est la méthode de décomposition par scénario la plus répandue dans la littérature. Cet algorithme correspond à une spécialisation de la méthode de Lagrangien augmenté conçue pour résoudre des programmes stochastiques généraux définis sur un arbre de scénarios. Pour résoudre $\mathcal{E}$ par cette méthode, on doit considérer la formulation mathématique équivalente suivante

$$(\tilde{\mathcal{E}}) \qquad \min_{x_t^\omega, \hat{x}_n \in \mathbb{R}_n} \sum_{\omega \in \Omega} \sum_{t=1}^{T} p_\omega g_t(x_t^\omega, \xi_t^\omega)$$

sous les contraintes

$$A_t^\omega x_t^\omega + B_t^\omega x_{t-1}^\omega = b_t^\omega \quad , \quad \forall t \in \{1, ..., T\},\ \omega \in \Omega, \tag{2.3}$$

$$x_t^\omega \in \mathcal{X}_t \quad , \quad \forall t \in \{1, ..., T\},\ \omega \in \Omega, \tag{2.4}$$

$$x_{t(n)}^\omega = \hat{x}_n \quad , \quad \forall n \in \mathcal{N}(\omega),\ \omega \in \Omega \qquad (\lambda_{n\omega}). \tag{2.5}$$

La formulation $\tilde{\mathcal{E}}$ est obtenue en appliquant les transformations suivantes à $\mathcal{E}$. Chaque vecteur de décision x_n défini à un noeud $n \in \mathcal{N}$ est remplacé par un nouveau vecteur de décision x_t^ω défini à la période t et au scénario ω correspondant. p_ω est la probabilité du scénario ω. ξ_t^ω est la réalisation à la période t du scénario ω. Les matrices de contraintes A_n et B_n sont remplacées par A_t^ω et B_t^ω, respectivement. L'ensemble $\mathcal{X}_t$ est défini par des contraintes statiques (p. ex. bilan énergétique). Les contraintes de non-anticipativité (CNA) (2.5) assurent les solutions réalisables sont invariantes par rapport aux scénario visitant à chaque noeud de l'arbre. L'ensemble $n \in \mathcal{N}(\omega)$ contient tous les noeuds visités par le scénario ω et par au moins un autre noeud. λ_n^ω représente les variables duales associées aux CNA.

L'APH classique fonctionne en appliquant une relaxation Lagrangienne augmentée aux CNA. Le Lagrangien augmenté à minimiser est donc

$$\mathcal{A}_\rho(x, \hat{x}, \lambda) = \sum_{\omega \in \Omega} p_\omega \sum_{t=1}^{T} g_t(x_t^\omega, \xi_t^\omega) + \sum_{\omega \in \Omega} \sum_{n \in \mathcal{N}(\omega)} \lambda'_{n\omega}(x_{t(n)}^\omega - \hat{x}_n) + \frac{1}{2}\rho \|x_{t(n)}^\omega - \hat{x}_n\|^2$$

où $x = (x_t^\omega)$ le vecteur de décision par période-scénario, $\hat{x} = (\hat{x}_n)$ est le vecteur de décision par noeud, $\lambda = (\lambda_{n\omega})$ est le vecteur dual, la constante $\rho > 0$ correspond au paramètre de pénalité et l'opérateur $\|\cdot\|$ représente la norme euclidienne.

À chaque itération k de l'APH, deux étapes sont effectuées. La première étape consiste

à minimiser $\mathcal{A}_{\rho_k}(x, \hat{x}_k, \lambda_k)$ en fixant l'approximation courante $\hat{x}_k$ et λ_k sous les contraintes physiques (2.3) et (2.4). Cette étape peut se faire efficacement en résolvant une série de sous-problèmes $\mathcal{S}_\omega$ définis sur les scénarios $\omega \in \Omega$. Chaque sous-problème correspond à un problème déterministe défini sur un scénario particulier de l'arbre auquel on a ajouté des termes linéaires et quadratiques pénalisant toute violation des CNA. Ceci fait de l'APH une méthode particulièrement bien adaptée pour les applications sur lesquelles un modèle stochastique doit être construit à partir d'un modèle déterministe existant. Dans bien des cas, cet avantage permet de réduire considérablement le temps de développement et de validation nécessaire au développement d'un modèle d'optimisation stochastique. La principale limitation de l'APH est causée par l'augmentation exponentielle du nombre de sous-problèmes et du nombre de CNA avec le niveau de branchement dans l'arbre. La deuxième étape consiste à mettre à jour la solution par noeuds $\hat{x}_k$ et les multiplicateurs de Lagrange λ_k. L'algorithme est décrit en détail au chapitre 4.

En général, la taille de chaque sous-problème augmente linéairement avec la taille du système considéré (nombre de réservoirs et centrales, réseau de transport). Conséquemment, cette méthode est applicable aux systèmes de haute dimension. Par contre, le nombre de sous-problèmes et le nombre de CNA augmentent exponentiellement avec le niveau de branchement dans l'arbre de scénarios. Pour ces systèmes, l'horizon de planification à moyen terme couvre typiquement plusieurs mois, voire quelques années et est discrétisé sur une base hebdomadaire. Conséquemment, un grand nombre de CNA doivent être satisfaites, ce qui peut augmenter le temps total de l'APH en augmentant le nombre d'itérations requis et en augmentant le niveau de difficulté des sous-problèmes. L'application de l'APH est particulièrement difficile lorsque les scénarios contenus dans l'arbre de scénarios sont très différents, ce qui a pour effet d'augmenter le niveau de difficulté de chaque CNA. Ce type de situation survient souvent dans certaines régions. Par exemple, au Canada, la variabilité saisonnière et interannuelle des apports hydriques naturels est particulièrement élevée et ceci peut faire augmenter considérablement la variabilité des solutions déterministes à chaque noeud.

L'applicabilité de l'APH a été démontrée sur une variété de problèmes d'optimisation incluant le design et à l'exploitation de réseaux (Mulvey et Vladimirou, 1991; Crainic *et al.*, 2011) et l'affectation de ressources (Watson et Woodruff, 2011). Ces problèmes contiennent généralement un faible nombre d'étapes et ont une structure mathématique très différente du PPMT que nous considérons dans cette thèse. Dans la littérature, les applications de l'APH en gestion de réservoirs hydroélectriques sont très rares et considèrent un horizon de courte portée avec un niveau de branchement modeste (2 mois, 6 périodes, une étape de

branchement). Conséquemment, le nombre de CNA à satisfaire est relativement faible comparativement au cas où un l'horizon temporel couvre plusieurs années. On peut difficilement évaluer à partir des études pasées si l'APH est applicable sur un horizon multiannuel avec un arbre contenant plusieurs étapes de branchement.

2.4 Analyse des différentes approches

Parmi les nombreuses méthodes explicites proposées dans la littérature, seulement un faible pourcentage d'entre elles est applicable à la gestion de grands systèmes hydroélectriques. Deux types d'approches sont traditionnellement utilisées pour faire la gestion de tels systèmes : l'approche par programmation dynamique duale stochatique et l'approche par arbre de scénarios. En général, aucun de ces deux types d'approches n'est supérieur à l'autre pour tous les types d'applications pratiques possibles. Le choix de l'approche à utiliser pour une application particulière dépend en grande partie du type de solution recherché par les gestionnaires. D'un côté, la programmation dynamique duale stochatique retourne une politique de gestion complète retournant une commande optimale à chaque période et pour n'importe quel état possible du système. Cette politique prend la forme d'une série des fonctions de Bellman convexes et linéaires par morceaux représentées par un nombre fini de coupes de Benders. D'un autre côté, les méthodes d'arbre de scénarios retournent un arbre de décision discret. L'arbre de décision possède une structure identique à l'arbre de scénarios et contient une commande optimale à chaque noeud. Dans bien des cas, seule la commande associée à la racine de l'arbre de décision est utilisée et cette solution est mise à jour régulièrement à mesure que de nouvelles informations sont rendues disponibles. La quantité de ressources nécessaires au développement et à la validation du méthode explicite est un autre facteur de décision à considérer lors du choix du type d'approche à utiliser. En général, l'implémentation informatique d'un modèle de programmation dynamique duale stochatique nécessite un investissement relativement important comparativement aux méthodes d'arbre de scénarios. Des versions commerciales de la programmation dynamique duale stochatique existent, mais ces implémentations sont souvent onéreuses et ne satisfont pas nécessairement les besoins particuliers de chaque entreprise.

CHAPITRE 3

DÉMARCHE ET ORGANISATION DE LA THÈSE

Dans cette thèse, nous considérons différentes méthodes pour résoudre le PPMT. Les méthodes considérées sont barbre de scénariosées sur une représentation par arbre de scénarios de l'incertitude. Trois différentes méthodes de décomposition sont utilisées pour résoudre le programme équivalent déterministe. La présente thèse est divisée en trois articles. Les chapitres 4, 5 et 6 correspondent au premier, deuxième et troisième article, respectivement. Chaque article est publié ou a été soumis dans un journal scientifique international.

3.1 Premier article

Cet article est accepté et publié dans le journal *Water Resources Research* (Carpentier *et al.*, 2013b). L'objectif de cet article consiste à évaluer numériquement l'applicabilité de l'APH pour faire la gestion de réservoirs hydroélectriques à forte capacité d'emmagarbre de scénariosinnement. Dans cet article, nous appliquons le schéma traditionnel de décomposition par scénario où chaque sous-problème est arbre de scénariossocié à un scénario particulier de l'arbre de scénarios. Pour la gestion de systèmes à haute capacité d'emmagarbre de scénariosinnement, l'horizon temporel couvre typiquement plusieurs dizaines voire quelques centaines de périodes et plusieurs étapes de branchement doivent être inclues dans l'arbre de scénarios. Ceci rend l'application de l'APH particulièrement difficile en raison du nombre élevé de CNA à satisfaire. En fait, le nombre élevé de termes linéaires et quadratiques ajoutés dans la fonction objectif peut ralentir considérablement le temps de calcul par itération de l'algorithme en augmentant le niveau de difficulté de chaque sous-problème. De plus, les solutions intermédiaires obtenues aux noeuds dupliqués sont susceptibles d'être très variables et ceci peut contribuer à ralentir substantiellement le taux de convergence de l'APH. Dans cette étude de carbre de scénarios, nous appliquons l'APH au PPMT auxquels font face les gestionnaires d'Hydro-Québec. L'horizon temporel considéré couvre un horizon de planification réparti sur 92 périodes avec un parbre de scénarios de temps hebdomadaire. Nous représentons l'incertitude hydrologique au moyen d'un arbre de scénarios contenant six étapes de branchement et 1635 noeuds.

3.2 Deuxième article

Le deuxième article a été soumis au journal *European Journal of Operational Research* (Carpentier *et al.*, 2013c). Dans cet article, nous considérons une version non conventionnelle de l'APH. Plutôt que d'appliquer le schéma de décomposition par scénario traditionnel, nous utilisons un schéma de décomposition multiscénario où chaque sous-problème est un PS défini sur un groupe de scénarios. Nous proposons une nouvelle heuristique visant à construire une partition optimale de l'ensemble de scénarios. La méthode proposée partitionne l'ensemble des scénarios de manière à minimiser le nombre de CNA devant être relaxées. Notre stratégie de partionnement permet de réduire le temps de calcul total de l'APH en accélératnt le taux de convergence et en réduisant le temps de calcul par itération. Cette amélioration s'explique en partie par la réduction du nombre termes de pénalité induite par la minimisation des CNA devant être relaxées. Aussi, le fait de minimiser le nombre de CNA devant être relaxées mène à des sous-arbres possédant un niveau de branchement plus élevé et ceci contribue à réduire la variabilité des solutions intermédiaires aux noeuds dupliqués.

3.3 Troisième article

Le troisième article a été soumis au journal *IEEE Transactions on Power Systems* (Carpentier *et al.*, 2013a). Dans cet article, nous présentons une version étendue de la méthode *L-Shaped* qui est bien adaptée pour faire la gestion de réservoirs saisonniers et multiannuels. La méthode proposée permet d'étendre le domaine d'applicabilité des méthodes de décomposition conventionnelles telles que l'APH ou la méthode *nested Benders* pour la gestion de tels systèmes. En fait, les méthodes de décomposition conventionnelles permettent de traiter des arbre de scénarios généraux et ceci permet, en principe, de prendre en compte les effets d'autocorrélation temporelle répartis sur nombre très élevé de périodes. Par exemple, cette représentation de l'incertitude permet théoriquement de représenter l'autocorrélation entre ξ_1 et ξ_T. La principale faiblesse des méthodes de décomposition conventionnelles est causée par l'augmentation exponentielle de la taille de l'arbre de scénarios avec le niveau de branchement utilisé. Lorsque T est élevé, ces méthodes ne sont applicables que si un faible niveau de branchement est utilisé, ce qui correspond typiquement à une approximation très grossière de la distribution de probabilité sous-jacente. Le but de notre méthode est de multiplier le niveau de branchement des arbre de scénarios pouvant être traitées avec ces méthodes. Notre méthode fonctionne en appliquant une décomposition de Benders à deux étape à $\mathcal{E}$. Nous posons l'hypothèse selon laquelle le processus stocharbre de scénariostique $\{\xi_t : t = 1, ..., T\}$ décrivant les paramètres aléatoires subit une perte de mémoire à la fin

de la période $\tau \in \{1, ..., T\}$. La période $\tau \gg 1$ correspond typiquement à un changement de régime. Cette hypothèse donne une structure particulière à l'arbre de scénarios que nous exploitons en utilisant une méthode de décomposition de Benders à deux étapes. Le problème maître et le sous-problème de deuxième étape sont des programmes stocharbre de scénarios-tiques définis sur un sous-arbre de scénarios. La méthode *L-Shaped* étendue est évaluée sur un problème de gestion à moyen terme réparti sur 104 semaines au Québec.

CHAPITRE 4

ARTICLE 1 : LONG-TERM MANAGEMENT OF A HYDROELECTRIC MULTIRESERVOIR SYSTEM UNDER UNCERTAINTY USING THE PROGRESSIVE HEDGING ALGORITHM

Pierre-Luc Carpentier[1,2], Michel Gendreau[1,2] and Fabian Bastin[1,3]

[1]Centre interuniversitaire de recherche sur les réseaux d'entreprise, la logistique et le transport (CIRRELT), Montréal, Québec, Canada.

[2]Department of Mathematics and Industrial Engineering, École Polytechnique de Montréal, Montrál, Québec, Canada.

[3]Department of Computer Science and Operations Research, University of Montreal, Montrál, Québec, Canada.

Publié le 27 mai 2013 dans Water Resources Research

Abstract

Among the numerous methods proposed over the past decades for solving reservoirs management problems, only a few are applicable on high dimensional reservoir systems (HDRSs). The progressive hedging algorithm (PHA) was rarely used for managing reservoir systems, but this method is a promising alternative to conventionally-used methods for managing HDRS (e.g. the stochastic dual dynamic programming). The PHA is especially well suited when a new stochastic optimization model must be built upon an existing deterministic optimization model (DOM). In such case, scenario subproblems can be resolved using an existing DOM with minor modifications. In previous studies, the PHA was rarely used and only tested on problems covering short-range planning horizons (2 months with 6 time periods) where a small number of non-anticipativity constraints (NACs) must be satisfied. Large reservoirs often need to be managed over a much longer planning horizon (1–5 years) containing many tens of time periods. In such case, convergence becomes much more difficult to achieve because of the larger number of NACs to be satisfied. Finding a non-anticipative solution

becomes particularly difficult when the input scenarios differ drastically. In this study, we demonstrate the applicability of the PHA for managing HDRSs over long-term (more than a year) horizons in highly uncertain decision environments. We apply the PHA on Hydro-Québec's reservoir system over a 92 weeks (periods) horizon. We analyze the performance of the PHA for different penalty parameter values. Deterministic solutions are compared to stochastic solution.

4.1 Introduction

Optimal operation of multipurpose reservoirs is among the most challenging and critical task facing water resources managers nowadays. Most reservoir systems fulfill numerous competing functions (e.g. power generation, flood/drought control, irrigation, navigation, recreation, ...) and tight operational constraints must be satisfied. Interconnected reservoirs are complex dynamical systems which often need to be managed over many tens of time periods under highly uncertain natural inflows. Hydroelectric reservoirs are particularly challenging to deal with since the power output of generating units is usually a nonlinear and non-convex function of head, turbined outflow and, sometimes, spillage. Furthermore, the operating range of these units is often characterized by forbidden zones due to low efficiency, cavitation or mechanical vibration effects. Hydroelectricity producers must also satisfy energy-related constraints (e.g. power demand and transmission network constraints), perform market transactions and coordinate hydroelectric generation with other production means (e.g. thermal plants, wind turbines, ...). Such producers are often affected by additional sources of uncertainty (e.g. power demand, market price, wind generation, ...). For example, Fleten et Kristoffersen (2008) takes into account prices and inflow uncertainty. Random vectors describing uncertain parameters at each time period usually have a continuous (joint) distribution and their dynamics is driven by a highly complex stochastic process (high dimensional, cross-correlated, large variability). Large hydroelectricity producers (e.g. in Canada, Norway and Brazil) operate high dimensional reservoir systems (HDRSs) with many tens of interconnected reservoirs. Without approximations, the mathematical problem to be solved is huge and has a highly complex mathematical structure (multistage, stochastic, nonlinear, non-convex, discrete decision variables).

In general, several simplifying assumptions must be made in optimization models (OMs) to ensure that the resulting mathematical program can be solved in reasonable time using available computing resources. Approximations also allow to reduce substantially the amount of time required to implement a new OM in the industry. In most models, hydroelectric gen-

eration units are aggregated as power plants and forbidden zones are omitted. Concave and piecewise linear generation functions are commonly used to represent the power output of hydro plants as a function of the water release and reservoir storage (Diniz et Maceira, 2008). Many OMs used operationally are built on the simplifying assumption that all input parameters are known with certainty over the entire horizon. Neglecting uncertainty in deterministic OMs (DOMs) reduces drastically the computational burden and simplifies the development process, but can lead to poor (high cost) solutions (Philbrick et Kitanidis, 1999). Stochastic OMs (SOMs) often have a superior performance level. In SOMs, the stochastic process describing continuous random parameters is generally assumed to possess a finite number of possible outcomes. In general, SOMs are computationally intensive and several years of development/validation time can be required before a new model can be used operationally. This is often a major obstacle for the application of stochastic models in real-world industrial applications. Despite the many years of academic research on RMPs, an important gap still exists between academic research and real-life applications as pointed out by Labadie (2004).

A rather simple approach to build a SOM would be to solve directly the so-called deterministic equivalent program using a readily available solver (e.g. GLPK, CPLEX, Gurobi, Xpress-MP, ...). Nowadays, these solvers enable to deal with problems containing millions of decision variables and constraints using regular PCs. However, this approach is rarely used in practice since the computational requirement (time, memory) for solving multistage stochastic programs grows very rapidly with the level of detail used to describe random parameters. Current capabilities of readily available solvers typically correspond to a very coarse representation of random parameters. Using a more detailed description of random parameters can be quite beneficial for systems operated in an highly uncertain decision environment. The vast majority of SOMs used for real-life applications are based on decomposition methods which allows to solve the large-scale stochastic reservoir management problems (RMPs) efficiently. Decomposition methods remains a very active area of research nowadays in the energy industry (Sagastizabal, 2012).

Over the past decades, several sophisticated decomposition methods were proposed in the literature for solving stochastic RMPs (Yeh, 1985; Wurbs, 1993; Labadie, 2004; Rani et Moreira, 2010). The most common approach relies on the dynamic programming (DP) principle (Bellman, 1957) and consists in applying time decomposition on the original problem. Instead of solving directly the (large) original stochastic program, the approach consists in solving a sequence of (small) subproblems. Each subproblem is associated to a specific time period of the planning horizon and a possible state of the dynamic system. DP-based

methods were applied repeatedly for solving RMPs in the literature (e.g. Yakowitz, 1982; Nandalal et Bogardi, 2007). The discrete stochastic DP (SDP) algorithm is a powerful tool for managing highly nonlinear and non-convex systems (e.g. Tejada-Guibert *et al.*, 1995), but this method is only applicable on low dimensional systems (less than 4 reservoirs) because of the *curse of dimensionality*. Due to this phenomenon, computational complexity of the SDP grows exponentially with the number of dimensions of state and control vectors. Several approximate DP (ADP) methods were developed to mitigate dimensionality problems in DP (Powell, 2011). One possible approach to reduce the size of state and control spaces is to aggregate many reservoirs into a larger hypothetical reservoir and apply SDP over the resulting system. This type of approach was applied on HDRSs by Archibald *et al.* (1997) and Turgeon (1998). The neuro-DP (NDP) algorithm (Bertsekas et Tsitsiklis, 1996) is a promising ADP approach to extend the applicability of discrete SDP. The state space dimensionality problem is mitigated by reducing the number of states for which subproblems are solved. Approximate values of the Bellman function for unsampled states are obtained by interpolation using an artificial neural network. Castelletti *et al.* (2007) applied the NDP algorithm on a three-reservoir system. Their results suggest this method is applicable for systems with more than three reservoirs. However, it is unlikely that this method could be applied on (HDRSs). Lee et Labadie (2007) and Castelletti *et al.* (2010) used a reinforcement learning method for solving a multireservoir operation problem. The stochastic dual DP (SDDP) algorithm proposed by Pereira et Pinto (1991) is another type of ADP method which can be used on HDRSs. With this method, the *curse of dimensionality* associated with state and control spaces is avoided since the method is based on continuous spaces. The SDDP was applied on first applied in Brazil (Maceira *et al.*, 2008), and, this method was applied subsequently on the Norwegian hydroelectric reservoir system in several past studies (e.g. Rotting et Gjelsvik, 1992; Gjelsvik *et al.*, 2010). Tilmant et Kelman (2007), Tilmant *et al.* (2008) and Goor *et al.* (2011) applied the SDDP on other HDRSs. The nested Benders decomposition algorithm is another approach which was applied on large reservoir systems by Jacobs *et al.* (1995) and Archibald *et al.* (1999). Another possible approach is the optimal reservoirs trajectories method proposed by Turgeon (2007).

The progressive hedging algorithm (PHA) proposed by Rockafellar et Wets (1991) was rarely used for solving RMPs in past studies. Nevertheless, this method is a promising alternative to time decomposition methods for managing HDRSs. With the PHA, the stochastic process describing uncertain input parameters are modeled by a finite scenario tree. Each scenario contained in the tree corresponds to a particular realization of the stochastic process. Contrary to conventionally-used methods, the PHA uses a scenario decomposition scheme

and is particularly well suited for industrial applications where a new SOM must be built upon an existing DOM. In such cases, scenario-subproblems can be solved directly using a lightly upgraded version of the existing DOM. Substantial amounts of development and validation time can be saved by upgrading the existing model rather than starting from scratch. Only simple penalty terms must added in the DOM's objective function to use it in the PHA. In fact, the PHA is an augmented lagrangian (AL) method in which non-anticipativity constraints (NACs) are relaxed. Penalty terms are used to enforce feasibility with respect to these constraints. The performance of the PHA can be greatly improved by solving scenario subproblems in parallel. Convergence of the PHA can become too slow to be applicable if the convergence rate is too low. Such phenomenon can happen if the number of NACs is too large or if scenarios differ too much. Indeed, the number of linear-quadratic penalty terms depends directly on the number of NACs to be satisfied. The required number of iterations to converge can also be quite sensitive to the penalty parameter choice. Tuning this parameter can be a time consuming task. Goncalves *et al.* (2011) applied successfully the PHA on the Brazilian system with stochastic inflows. These authors considered a relatively short planning horizon covering 2 months with 6 time periods. In practice, some RMPs need to be distributed over a much longer planning horizon and much more branching nodes are required to represent hydrologic uncertainty adequately. It is impossible, only from this study, to determine if the PHA is still applicable over these larger problems and on different hydro-climatic conditions (e.g. in snow-dominated watersheds).

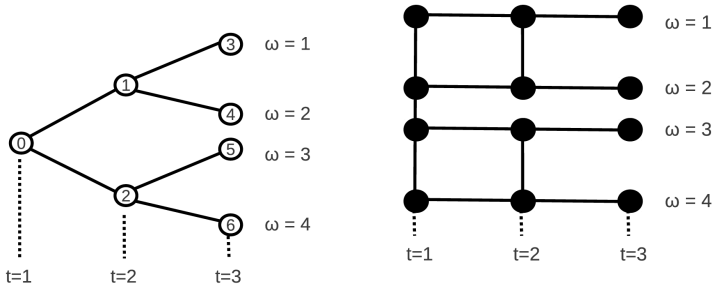

Figure 4.1 Example of node-wise (left) and scenario-wise (right) representations of a scenario tree.

In this paper, we demonstrate that the PHA can be used solve efficiently large RMPs distributed over an extended planning horizons under highly uncertain operating conditions. To achieve this, we perform a case study based on a simplified version of Hydro-Québec's RMP. This problem is particularly challenging to solve using the PHA given the size and storage capacity of the reservoir system to be managed and the variability of hydro-climatic condi-

tions. Operating conditions are highly variable throughout the year and tight operational constraints have to be satisfied. A 92 periods horizon with weekly time steps is considered in this experiment. In this problem, hydrologic uncertainty is huge, especially during the spring flood season. Natural inflows uncertainty is modeled by a scenario tree containing drastically different inflow scenarios constructed from the historical record. The scenario tree was designed to include the most difficult problem to deal with the PHA. Also, transmission network constraints are taken into account in this case study. These constraints are rarely considered in the water resources literature.

The remainder of the paper is organized as follow. The mathematical formulation of the optimization problem is presented in Section 4.2. Section 4.3 presents the solution method in details. Section 4.4 describes a case study based on Hydro-Québec's power system. Numerical results are presented in Section 4.5. Comments and conclusions are drawn in Section 4.6.

4.2 Problem formulation

4.2.1 Stochastic program

We consider a general mathematical formulation for the stochastic RMP which can be applied on different multicomponent reservoir systems affected by various sources of uncertainty. For example, the system operated may contain cascaded reservoirs, hydroelectric power plants, a transmission network, thermal plants or other components. Market transactions can be part of the controlled decisions. Several external parameters can be uncertain such natural inflows, power/water demand, intermittent production means (e.g. wind turbines, isolated run-of-the-river plants) or market prices. The optimization problem is formulated as the following multistage stochastic program:

$$(\mathcal{P}) \qquad \max \mathbb{E}\left[\sum_{t=1}^{T} g_t(x_t, \xi_t)\right] \tag{4.1}$$

subject to

$$A_t x_t + B_t x_{t-1} = d_t(\xi_t), \quad \forall t = 1, ..., T \tag{4.2}$$

$$C_t x_t + D_t x_{t-1} \leq c_t(\xi_t), \qquad \forall t \in \mathcal{T} \tag{4.3}$$

$$\underline{x}_t \leq x_t \leq x_t, \quad \forall t = 1, ..., T \tag{4.4}$$

$$x_t \in \mathcal{F}_t, \quad \forall t = 1, ..., T. \tag{4.5}$$

The objective function (4.1) to be maximized is the expected total benefit obtained from

operating a multicomponent reservoir system at time periods $t = 1, ..., T$. The mathematical expectation operator $\mathbb{E}[\cdot]$ is computed with respect to the probability distribution of random vectors ξ_t whose components represent the problem's uncertain input parameter at the corresponding time period t. The benefit g_t at time period t depends on random vector ξ_t and are concave function of decision vector x_t. Each components of x_t corresponds to a decision variables at time period t (e.g. turbined outflow, spillage, reservoir storage, thermal generation, transmission network flow, market transactions, etc.). Linear constraints (4.2)–(4.4) represent physical constraints (e.g. energy/water balance, hydro generation, thermal generation, etc.). Matrices A_t, B_t, C_t and D_t have known coefficients. Coefficients of right-hand size $c_t(\xi_t)$ and $d_t(\xi_t)$ represent external parameters some which may be random (e.g. power/water demand, inflows).

In this problem, we assume that the realization of ξ_t is revealed to the decision maker before x_t is chosen. On the other hand, we assume that the decision maker cannot anticipate the realization of future random vectors $\xi_{t+1}, \xi_{t+2}, ..., \xi_T$ when x_t is chosen. We ensure that x_t can only be adapted to known informations $(\xi_1, ...\xi_t)$ by non-anticipativity constraints (NACs) (4.5). Each set $\mathcal{F}_t$ contains all non-anticipative decision vectors x_t at the beginning of time period t. Any non-anticipative solution can only depend on the available informations at the beginning of time period t.

4.2.2 Scenario tree

We assume that random vectors ξ_t at time periods $t = 1, ..., T$ are discretely distributed and that their joint distribution has a finite number of possible outcomes. This assumption enables us to represent the stochastic process $\{\xi_t : t = 1, ..., T\}$ using a finite scenario tree described by a set of nodes $\mathcal{N}$. Scenario trees is a powerful approach for modeling uncertainty in multistage stochastic programs. This approach allows to represent stochastic processes possessing a very complex space- and time-autocorrelation structure and can be applied on various type of uncertain parameters. Scenario trees can be used to represent natural inflows uncertainty in various hydro-climatic conditions. Other random parameters (e.g. load, prices) can also be represented by this approach.

Each scenario (realization) $\omega \in \Omega$ of the stochastic process is a sequence $\{\xi_{t\omega} : t = 1, ..., T\}$ with probability of $p_{\ell(\omega)} \in (0, 1)$. Each vector $\xi_{t\omega}$ is the realization of random vector ξ_t at time period t for scenario ω. Each pair (ω, t) of scenario ω and time period t is associated to a particular tree node n. A simple example of a scenario tree with two branching stages and

$\mathcal{N} = \{0, 1, 2, 3, 4, 5, 6\}$ is presented in Figure 4.1. In this example, scenario $\omega = 1$ corresponds to the path 0-1-3.

4.2.3 Deterministic equivalent program

The stochastic program $\mathcal{P}$ can be written explicitly as the following deterministic equivalent linear program:

$$(\mathcal{E}) \ \max \sum_{\omega \in \Omega} p_{\ell(\omega)} \sum_{t=1,\ldots,T} g_t(x_{t\omega}, \xi_{t\omega}) \tag{4.6}$$

subject to

$$x_\omega \in X_\omega, \qquad \forall \omega \in \Omega \tag{4.7}$$

$$x_{t(n)\omega} - \hat{x}_n = 0, \quad \forall n \in \mathcal{N}_*, \omega \in \Omega(n) \quad (\lambda_{n\omega}) \tag{4.8}$$

where $x_{t\omega}$ is the decision vector at time period t for scenario ω. Decision vector $x_\omega = (x_{t\omega})$ contains the decisions at all time periods for scenario ω. Each polyhedral set X_ω is defined by physical constraints (4.2)–(4.4) associated with inflow scenario ω. The linear equality constraints (4.8) are defined at tree nodes $n \in \mathcal{N}_*$ which are visited by more than one scenario. These constraints ensure that control vectors $x_{t\omega}$ are scenario invariant for all scenarios $\omega \in \Omega(n) \subseteq \Omega$ visiting node $n \in \mathcal{N}_* \subset \mathcal{N}$. $\hat{x}_n$ is the decision vector associated with tree node n. $\lambda_{n\omega}$ is the dual vector for NACs associated to node n for scenario ω. On Figure 4.1, we have $\mathcal{N}_* = \{0, 1, 2\}$, $\Omega(0) = \{1, 2, 3, 4\}$, $\Omega(1) = \{1, 2\}$ and $\Omega(2) = \{3, 4\}$.

4.3 Solution method

4.3.1 Scenario decomposition

The constraint matrix of $\mathcal{E}$ is sparse and exhibits a special structure which can be exploited efficiently using a scenario decomposition method. As shown in Figure 4.2, physical constraints (4.2)–(4.4) corresponds to small separate blocks containing nonzero coefficients in the constraints matrix. Each of these blocks is associated to a particular scenario $\omega \in \Omega$ in the tree. NACs (4.8) corresponds to a large block with nonzero coefficients. Remaining matrix coefficients are equal to zero. Physical constraints are relatively easy to deal with since they correspond to a set of independent blocks in the constraint matrix. Conversely, NACs are more difficult to deal with since they couple scenario blocks to one another.

In order to solve $\mathcal{E}$ using the PHA, we apply Lagrangean relaxation on NACs and penalize

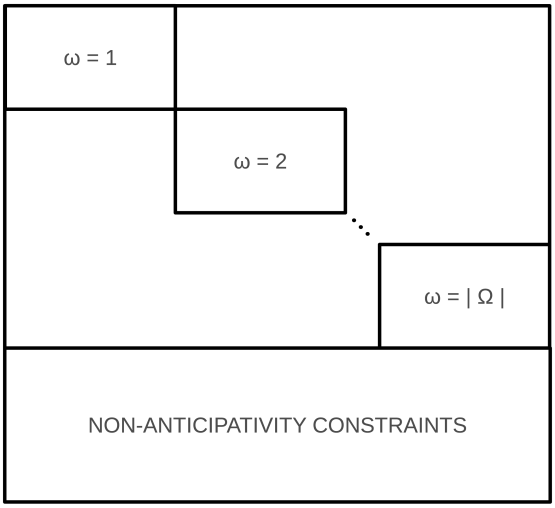

Figure 4.2 Structure of the constraints matrix.

quadratically any violation of it. The resulting augmented Lagrangian is

$$\mathcal{A}_\rho(x,\hat{x},\lambda) = \sum_{\omega\in\Omega} p_{\ell(\omega)} \sum_{t=1}^{T} g_t(x_{t\omega}) - \sum_{\omega\in\Omega}\sum_{n\in\mathcal{N}(\omega)} \left(\lambda'_{n\omega}(x_{t(n)\omega} - \hat{x}_n) + \frac{\rho}{2}\|x_{t(n)\omega} - \hat{x}_n\|^2\right)$$

where $\rho > 0$ is the penalty parameter, $x = (x_{t\omega})$ is the scenario-wise solution, $\hat{x} = (\hat{x}_n)$ is the node-wise solution, $\lambda = (\lambda_{n\omega})$ is the dual vector associated with NACs and $\mathcal{N}(\omega)$ contains nodes visited by scenario ω. All vectors are column-vectors, the operator $(\cdot)'$ represents the transpose and $\|\cdot\|$ is the Euclidean norm.

4.3.2 Progressive hedging algorithm

The PHA is initialized with an estimation of $\hat{x}^0$ and λ^0. At each iteration $k = 0, 1, 2, \ldots$, two steps are performed:

Step 1. Find a new scenario-wise solution $x^{k+1} = (x^{k+1}_{t\omega})$ by maximizing $\mathcal{A}_{\rho_k}(x, \hat{x}^k, \lambda^k)$ for $x \in X$ using the current $\hat{x}^k$ and λ^k. The set X is defined by physical constraints (4.7) for all scenarios $\omega \in \Omega$. This large optimization problem can be decomposed into much smaller scenario subproblems

$$(\mathcal{S}^k_\omega) \max_{x_\omega\in X_\omega} p_{\ell(\omega)} \sum_{t=1,\ldots,T} g_t(x_{t\omega}) - \sum_{n\in\mathcal{N}_*(\omega)} \left((\lambda^k_{n\omega})'(x_{t(n)\omega} - \hat{x}^k_n) + \frac{\rho}{2}\|x_{t(n)\omega} - \hat{x}^k_n\|^2\right).$$

where $\mathcal{N}_*(\omega) \subseteq \mathcal{N}_*$ is the set of nodes visited by scenario ω. Each subproblem $\mathcal{S}_\omega^k$ corresponds to a deterministic version of $\mathcal{E}$ for scenario ω with linear and quadratic penalty terms. Subproblems can be solved sequentially or in parallel.

Step 2. Update node-wise control vectors by averaging scenario-wise control vectors using

$$\hat{x}_n^{k+1} \leftarrow \sum_{\omega \in \Omega(n)} p_{\ell(\omega)} x_{t(n)\omega}^{k+1} / \sum_{\omega \in \Omega(n)} p_{\ell(\omega)}, \qquad \forall n \in \mathcal{N}_*.$$

Update dual vectors

$$\lambda_{n\omega}^{k+1} \leftarrow \lambda_{n\omega}^k + \rho \left(x_{t(n)\omega}^{k+1} - \hat{x}_n^{k+1} \right), \qquad \forall n \in \mathcal{N}_*, \omega \in \Omega(n).$$

Verify if the two following stopping conditions

$$\zeta_k := \frac{1}{T} \sum_{\omega \in \Omega} p_{\ell(\omega)} \sum_{t=1,\ldots,T} \| x_{t\omega}^{k+1} - \hat{x}_{n(t,\omega)}^{k+1} \|^2 < \hat{\epsilon}_0 \tag{4.9}$$

$$\chi_k := \frac{\left| \mathcal{A}_\rho(x^{k+1}, \hat{x}^{k+1}, \lambda^{k+1}) - \mathcal{A}_\rho(x^k, \hat{x}^k, \lambda^k) \right|}{\mathcal{A}_\rho(x^k, \hat{x}^k, \lambda^k)} < \hat{\epsilon}_1 \tag{4.10}$$

are satisfied for some stopping criteria $\hat{\epsilon}_0$, $\hat{\epsilon}_1 > 0$. Parameters ζ_k and χ_k measure violation of NACs and the relative improvement of the augmented Lagrangian at the current iteration k. If the solution does not satisfy conditions (4.9) and (4.10), return to step 1.

Physical constraints are satisfied at each iteration since they are treated directly in $\mathcal{S}_\omega^k$. However, NACs are expected to be violated in early iterations since these constraints are treated indirectly by penalizing violations. Linear and quadratic penalty terms in $\mathcal{S}_\omega^k$ will ensure that the violation of NACs will decrease gradually through the iterative process in order to obtain a feasible (non-anticipative) solution. The PHA is an exact method for $\mathcal{E}$ since this problem is linear (and convex). A proof of convergence is presented in Rockafellar et Wets (1991).

4.4 Case study

Given that the aim of this study is to demonstrate the applicability of the PHA for solving long-term RMPs, we apply this method on a simplified version of Hydro-Québec's RMP (HQRMP) with stochastic inflows. This problem covers a long-term planning horizon and is

a particular case of the general stochastic program $\mathcal{E}$. Hydro-Québec is the largest hydroelectricity producer in Canada (Fortin, 2008). The company relies almost exclusively on hydro plants with multiannual storage capacity to produce electricity. Most of Hydro-Québec's production is sold at a constant price on Québec's regulated market. Energy can be sold or purchased on deregulated markets located outside of the province.

The HQRMP is solved operationally by the company's system managers to find optimal water release targets for the current week and reservoir state trajectories for the upcoming 1.5–2 years. Weekly water release targets are used as an input parameter in a highly detailed short-term unit commitment model. System managers update these targets on a weekly basis by running a DOM with three different natural inflows scenarios (e.g. wet, dry, average) using the newly observed reservoir state (level). The aim of the HQRMP is to meet a time-varying electrical load pattern exactly using only controllable hydro plants over a 78–104 periods horizon with weekly discretization.

Neglecting uncertainty in the DOM simplifies computationally the problem to be solved, but this assumption yields to overly optimistic solutions which must be used very carefully in practice. Deterministic solutions are usually quite sensitive to which scenario is used in the model making them difficult to interpret. An optimal solution for one scenario can have a poor performance (spillage, low head/efficiency) if it is used on a different hydrological scenario.

The PHA approach is particularly well suited since the company already possess a DOM which is trusted by system managers for handling head and efficiency variations. The PHA-based optimization model described in this section is an extension of the company's current DOM. Scenario-subproblems are solved using a streamlined version of the company's DOM to which linear-quadratic penalty terms were added for penalizing violation of NACs. It is worth mentioning that the PHA model could easily be extended for other contexts where market transactions and controllable non hydraulic generation means (e.g. thermal plants) would be part of the optimization problem. If necessary, additional sources of uncertainty (e.g. market prices, power/water demand, wind turbines, isolated run-of-the-river plants) could also be included in the random vectors ξ_t.

4.4.1 Hydroelectric reservoir sytem

The only controllable production mean considered in the HQRMP is hydroelectricity. Intermittent run-of-the-river plants (without storage capacity), non-hydraulic production means (e.g. wind turbines and thermal plants) and market transactions are considered as predetermined energy sources at this stage of Hydro-Québec's decision process. Controllable hydro plants $i \in I$ can release water to produce electricity or spill it without production from an upstream reservoir $j(i) \in J$. We assume that generating unit efficiency is a monotonically decreasing function of the turbined outflow at each plant. Forbidden zones are not taken into account. The power output and maximal turbined outflow are head sensitive. Head of hydro plant i depends on the water level in the upstream reservoir $j(i)$ and, in some cases, may also depend on the water level of reservoir $\delta(i)$ located directly downstream of it. This situation can happen when reservoir $j(i) \in J$ is emptied directly into reservoir $\delta(i) \in J$. The set $I(j)$ contains all hydro plants whose upstream reservoir is j. We assume that all hydro plants $i \in I(j)$ are mutually independent. In other word, for a given reservoir j, the production of any plant $i_1 \in I(j)$ cannot affect the production of any other plant $i_2 \in I(j)$. Each set $\mathcal{U}(j)$ contains all reservoirs located directly upstream from reservoir j. We assume that water travels within one time period (week) from any reservoir $u \in \mathcal{U}(j)$ to j.

For sake of simplicity, we assume no maintenance is scheduled on generating units and that the power output of each hydro plant i is returned by a three-dimensional concave and piecewise linear function $\phi_i(q_i, v_{j(i)}, v_{\delta(i)})$ (MW) which depend on the instantaneous turbined outflow q_i ($\mathrm{m}^3\ \mathrm{s}^{-1}$) and on the storage $v_{j(i)}$ and $v_{\delta(i)}$ in the upstream reservoir $j(i)$ and downstream reservoir $\delta(i)$, respectively. Maintenance could easily be included by using a different generation function for each time period. We assume that the maximal turbined outflow $q_i^{\max}$ increases linearly with head as follow

$$q_i^{\max} = \alpha_0^i + \alpha_1^i v_{j(i)} + \alpha_2^i v_{\delta(i)}$$

where α_0^i, α_1^i and α_2^i are known constants. Figure 4.3 shows an example of hydroelectric generation function with three pieces. The lower and upper curves represents the power output of an hydro plant operated at low and high head, respectively.

In this experiment, we consider the reservoir system shown on Figures 4.4–4.7 which accounts for most of Hydro-Québec's storage and hydroelectric generation capacity. The system is organized in 6 different subsystems and contains 21 reservoirs. The total storage capacity is 189,414 hm^3. Reservoirs are supplying water to 25 hydroelectric power plants

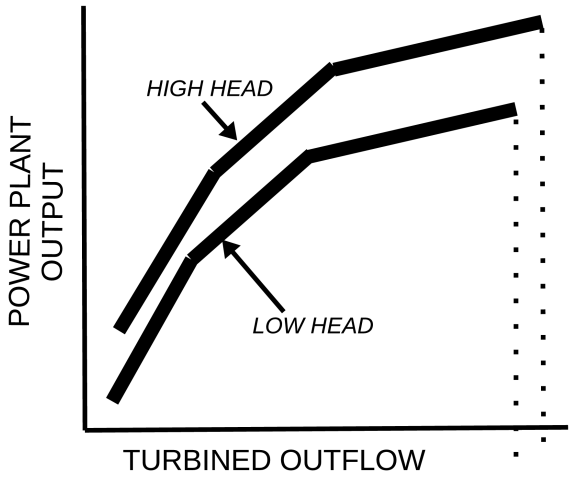

Figure 4.3 Power output of a variable-head hydro plant.

which possess a total installed capacity of 33.2 GW. Generation functions for each hydro plant is described using three hyperplanes. The initial reservoir storage is 118,685 hm^3. Tables 4.4.1 and 4.4.1 show the characteristics of each reservoir and hydro plant, respectively.

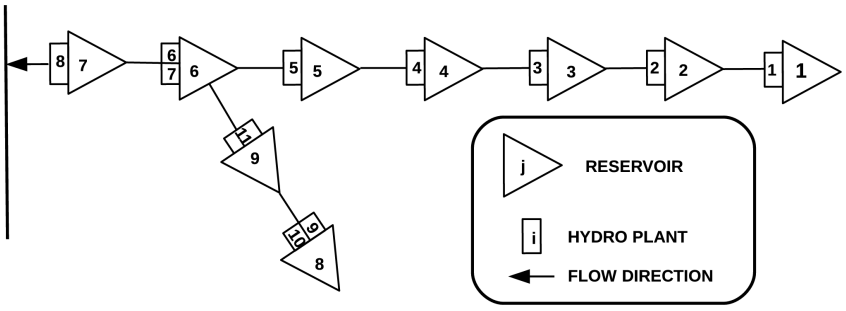

Figure 4.4 Hydroelectric system in zone $z = 1$.

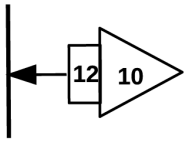

Figure 4.5 Hydroelectric system in zone $z = 2$.

Table 4.1 Characteristics of hydro plants.

i	Installed capacity (MW)	Head (m)	Number of units
1	469	37.5	2
2	319	27.4	2
3	878	57.3	6
4	2,779	116.7	9
5	2,417	79	12
6	5,616	137.16	16
7	2,106	138.5	6
8	1,436	27.5	12
9	507	63	3
10	812	63	3
11	127	11	3
12	5,428	318	11
13	884	330	2
14	1,596	141.8	8
15	1,064	144.5	4
16	1,244	94.19	6
17	1,145	70.11	8
18	184	36.58	3
19	235	37.8	7
20	526	152	2
21	785	120.55	4
22	1,026	143.57	4
23	523	82.3	3
24	1,178	266.7	8
25	869	115.83	5

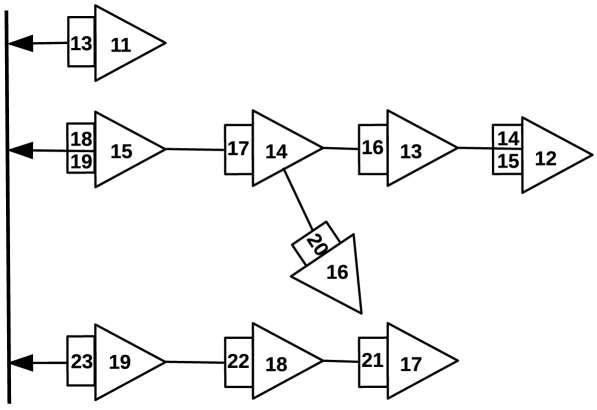

Figure 4.6 Hydroelectric system in zone $z = 3$.

Table 4.2 Characteristics of reservoirs.

j	Maximal storage (hm^3)
1	39,008
2	338
3	5,820
4	7,079
5	25,195
6	19,369
7	85
8	4,211
9	3,393
10	32,499
11	3,279
12	28,224
13	209
14	111
15	12
16	2,436
17	10,940
18	10
19	9
20	7,063
21	124

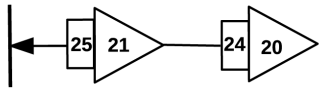

Figure 4.7 Hydroelectric system in zone $z = 4$.

4.4.2 Transmission network

Each hydro plants i is connected to a particular zone $z(i) \in Z$ of an highly simplified transmission network $G = (Z, L)$. Even though it the representation of G is simple, it is adequate for this application since the only purpose of the HQRMP is to schedule reservoir release targets over a multiannual planning horizon with weekly discretization. This model is not used to determine the optimal power flow on the transmission network. These release targets are used as an input in a highly detailed short-term model. Links $\ell = (z_1, z_2) \in L$ allow to transfer energy from zone z_1 to z_2. Sets $L_+(z)$ and $L_-(z)$ contain transmission links entering and outgoing zone z, respectively. The power flow X_ℓ (MW) entering each link ℓ is a controlled variable. The power outgoing link ℓ is $(1 - \epsilon_\ell)X$ where $\epsilon_\ell \in [0, 1]$ represents the fraction of energy lost on link ℓ. In general, this parameter depends on the line length and other physical characteristics. Each zone $z \in Z$ of the transmission network is supplied by hydro plants $i \in I(z)$.

We consider the transmission network shown on Figure 4.8 which contains 5 zones and 6 links. Approximately half of the total storage and installed capacity is in zone $z = 1$ shown on Figure 4.4. About 43% of theses quantities is located in zones $z = 2$ and $z = 3$ which are shown on Figures 4.5 and 4.6, respectively. The remaining storage and installed capacity is located in zone $z = 4$.

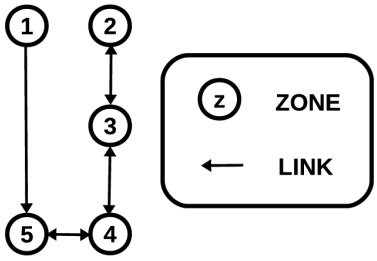

Figure 4.8 Transmission network.

4.4.3 Electrical load

The power system must balance exactly the net electrical load d^{net}_{zct} (MW) at all time periods $t = 1, ..., T$ and in each zone $z \in Z$ of the transmission network. Short-term (hourly, daily) load variations within any given time period t are represented using a discretized load duration curve. Each load level $c \in C$ has an associated duration Δt_c (h). Figure 4.9 shows

a simple example of weekly load duration curve where $C = \{1, 2, 3\}$, $\Delta t_1 = \Delta t_3 = 20$ hours and $\Delta t_2 = 128$ hours.

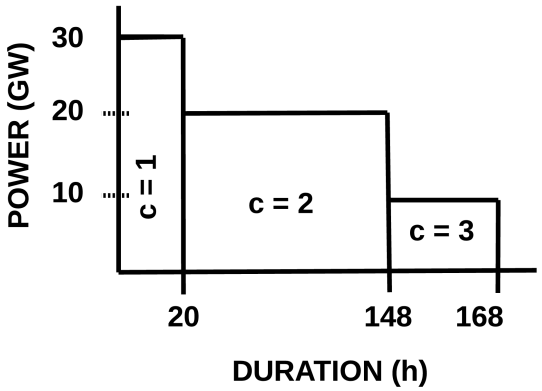

Figure 4.9 Example of load duration curve.

In the HQRMP, the net electrical load is computed as follow:

$$d_{zct}^{net} := d_{zct}^{prov} - Pred_{zct} + Sales_{zct} - Purchases_{zct}$$

where d_{zct}^{prov} (MW) is the provincial power demand, $Pred_{zct}$ (MW) is the predetermined generation (e.g. wind generation, thermal, run-of-the-river, market, ...), $Purchases_{zct}$ (MW) is the market purchases and $Sales_{zct}$ is the market sales. We assume that parameters d_{zct}^{prov}, $Pred_{zct}$, $Sales_{zct}$ and $Purchases_{zct}$ are known over the entire planning horizon.

In this experiment, we consider a $T = 92$ periods planning horizon which begins February 1^{st} during the peak load period. The end of last time period corresponds to October 31^{st} of the horizon's second year. The total electrical load to be satisfied is 282.6 TWh. Short-term (hourly, daily) variations of load at each period are represented with three levels. Figure 4.10 shows variations of load intensity over the entire horizon for each level. The load is very high during winter and decreases importantly during summer and fall. The energy load is distributed in zones 2, 3, 4 and 5 of the transmission network as follow. 65% of the total load is located in zone $z = 5$. 19% of the total load located in zone $z = 4$. The remaining load (16%) is located in zones $z = 2$ and $z = 3$.

4.4.4 Inflow scenario tree

In this case study, we consider the most challenging situation facing Hydro-Québec system managers which occurs a several weeks before the spring flood whose timing and intensity is highly unpredictable. We assume that natural inflows is the only source of uncertainty. Other sources of uncertainty also exist, but are much less important. Hydrologic uncertainty

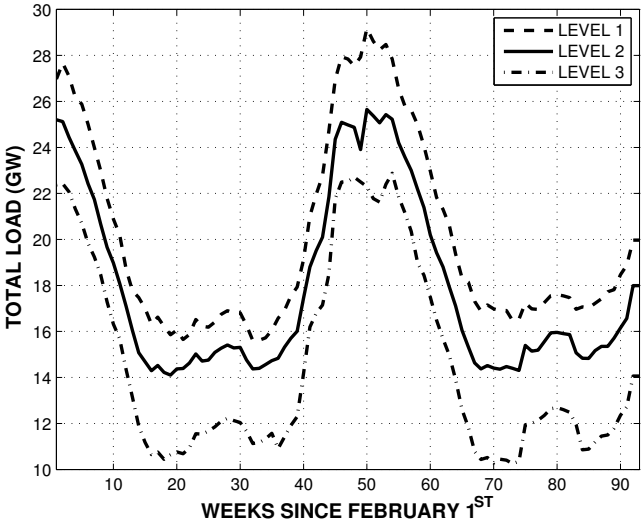

Figure 4.10 Electrical load.

is represented by a scenario tree. Any other hydro-climatic condition (e.g. during summer) would involve less uncertainty and would be easier to deal with. Therefore, we focus on spring flood management. Representing precisely the probability distributions of natural inflows is a complex task in itself and this topic is not the main objective of this study. Given that scenario tree construction is well beyond the scope of the current paper, we consider the simplified scenario tree shown on Figure 4.11 which contains six branching stages and 1,635 nodes. To make the HQRMP as challenging as possible, we constructed 64 drastically different scenarios. For sake of simplicity, we assume that the occurrence probability is the same for all scenarios. Therefore, $p_{\ell(\omega)} = 1/64$ for all $\omega \in \Omega$. The planning horizon covers two successive hydrological years. Figure 4.12 and 4.13 show details of the branching structure at the first and second year, respectively. Each year has three branching stages and contains 15 segments. An inflow vector $\mathcal{I}_n = (\mathcal{I}_{nj})$ (hm^3) is defined at each tree node n. Each component $\mathcal{I}_{nj}$ (hm^3) represents the volume of inflow in reservoir j. The sequence of inflow vectors on any given segment of nodes corresponds to the natural inflow observed during a particular year of the historical record. Figure 4.14 shows the historical data of total inflow for all reservoirs.

Periods $t = 1, ..., 12$ corresponds to the first winter season during which natural inflows is very weak and identical for all scenarios. Average winter conditions are used to represent hydrological conditions on the segment 0–11. The second section ($t = 13, ..., 26$) corresponds to the spring flood season. The segment 11–14 corresponds to the earliest spring flood onset

which occurs at week $t = 12$. The segment 11–20 corresponds to persistent freezing season with a late onset of the spring flood ($t = 15$). The segments 14–31 and 20–50 represents the case where the inflow intensity is highest on the historical record. Conversely, the segments 14–42 and 20–58 represents the case where the inflow intensity is lowest on the historical record. The summer and fall seasons cover time periods $t = 26, ..., 40$ and are represented by one branching stage. Segments 31–72, 42–100, 50–128 and 58–156 represent the wettest conditions on record. Conversely, segments 31–86, 42–114, 50–142 and 58–170 represent the driest conditions on record. The second year covers time periods $t = 41, ..., 92$ and possess a very similar structure to year 1 as shown on 4.13. The only difference lies in the fact that the winter season covers more time periods.

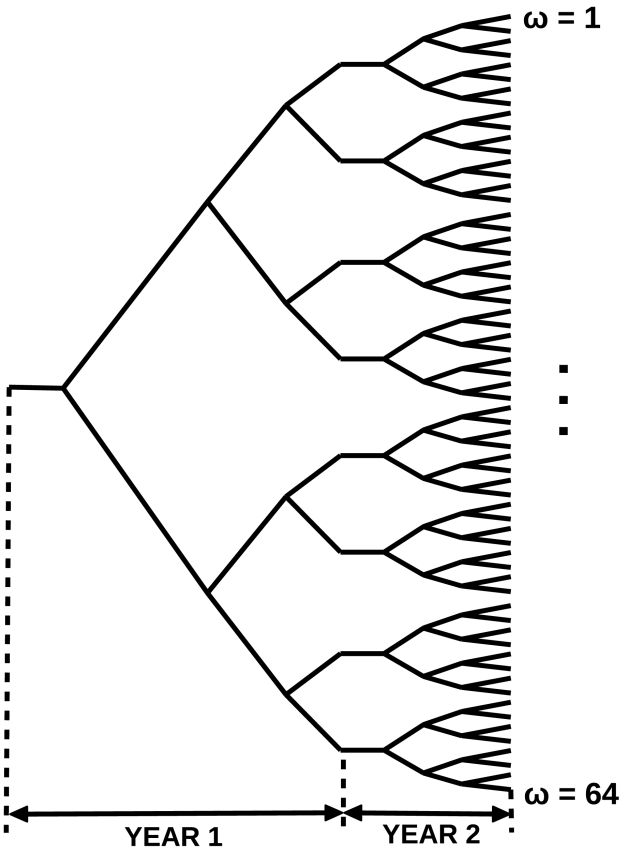

Figure 4.11 Scenario tree.

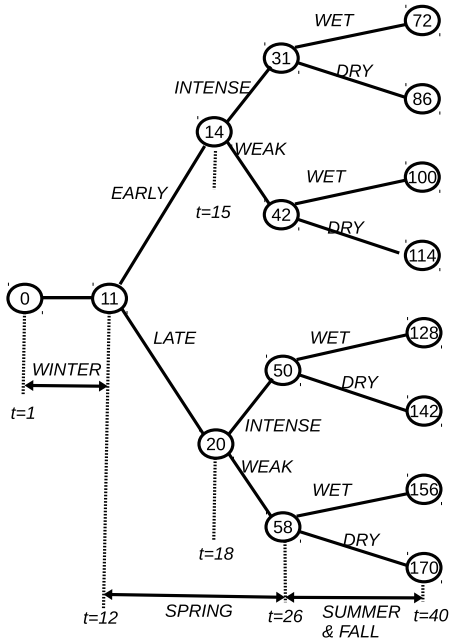

Figure 4.12 Branching structure at the first year.

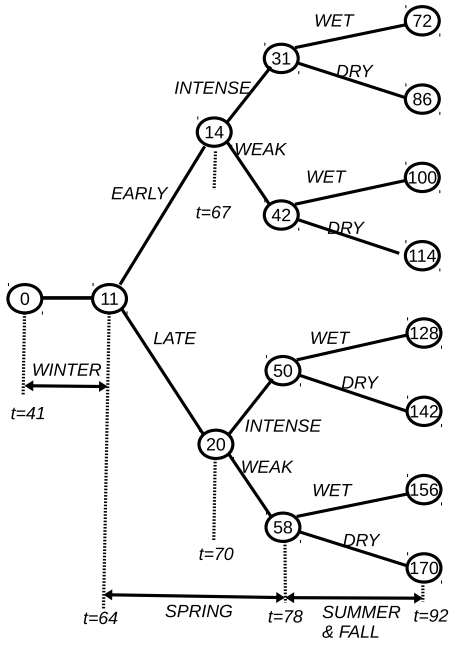

Figure 4.13 Branching structure at the second year.

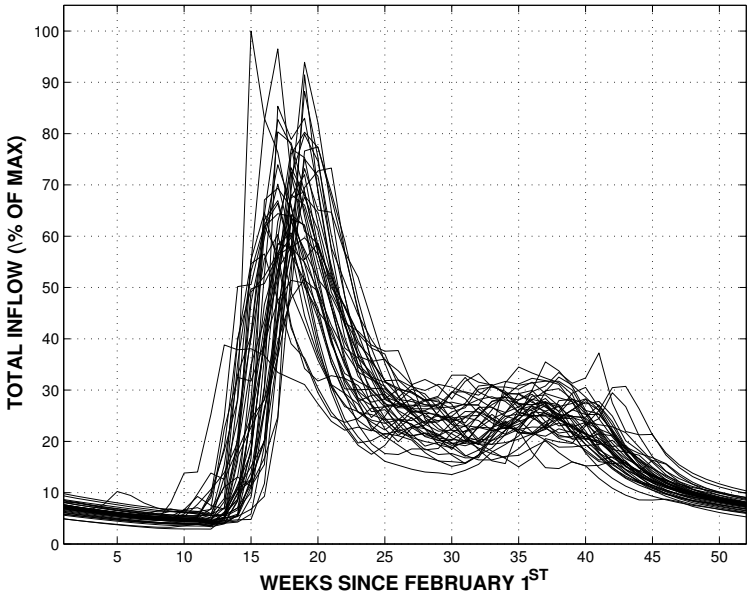

Figure 4.14 Seasonal cycle of total reservoir inflows for 42 years of historical record beginning on Februray 1^{st}.

4.4.5 Stochastic program formulation

Objective function

The scheduling of power generation and energy trading is not performed at the same stage of the Hydro-Québec's decision process. In practice, the power generation of non-hydraulic production means (e.g. thermal plants) and market transactions are decided before solving the HQRMP. Therefore, these decisions are treated as input parameters for the HQRMP. In this problem, only cost-free hydro plants can be controlled to meet a given load pattern. At this stage of the decision process, the system's performance depends essentially on head and hydroelectric efficiency are managed and on how desirable is the final system state at the end of the planning horizon. The decision maker must use as little water as possible to meet the domestic load over the entire planning horizon. Head and unit efficiency variations and spillage have a direct impact on how efficiently water is used. Also, the system's final state should be adequate for meeting the load beyond the end of the planning horizon ($t = T$). The current performance measure used operationally at Hydro-Quebec depends only on the final state of the system and the objective function to be maximized

$$\max \sum_{\omega \in \Omega} p_{\ell(\omega)} \sum_{j \in J} \mathcal{R}_j(v_{jT}^{\omega}) \tag{4.11}$$

is the expected total economic value of water at the end of the planning horizon. It can be shown that equation (4.11) is a particular case of equation (4.6). No cost or revenue are produced at time periods $t = 1, 2, ..., T-1$. Therefore, the revenue function $g_t(x_t^\omega) = 0$ for $t = 1, 2, ..., T-1$. The revenue function at the last time period ($t = T$) is $g_T(x_{T\omega}, \xi_{T\omega}) = \sum_{j \in J} \mathcal{R}_j(v_{jT}^\omega)$ and depends only on the volume of water v_{jT}^ω (hm^3) stored in reservoirs $j \in J$ at the end of time period T for scenarios ω. The economic value of water $\mathcal{R}_j(v_{jT}^\omega)$ ($) in reservoir j is a monotonically increasing, concave and piecewise linear function v_{jT}^ω. These functions are tuned by Hydro-Québec system managers using a simplified DP-based long-term optimization model which takes into account the energy prices and operational constraints beyond the planning horizon. The slope decrease as the storage approaches its upper limit, the slope of $\mathcal{R}_j$ decreases because of increased spillage probability. Because it is increasing, any optimal solution will use as little water as possible. Reward functions R_j are described using 7 pieces. In the optimization model, the concave and piecewise linear relationship $\mathcal{R}_j(v_{jT}^\omega)$ can be represented using a finite number of cuts $h \in \mathcal{H}_{Rj}$ which correspond to linear inequality constraints

$$\mathcal{R}_j^\omega \leq \eta_{jh}^0 + \eta_{jh}^1 v_{jT}^\omega, \qquad \forall j, h, \omega. \tag{4.12}$$

Here, η_{jh}^0 ($) and η_{jh}^1 ($/hm^3) represent the parameters of cut h for reservoir j. Fig 4.15 shows a simple example where $\mathcal{H}_R = \{1, 2, 3\}$. The vertical axis represents the reward function $R_j(v_j)$ and the horizontal axis corresponds to the volume of water stored in reservoir j.

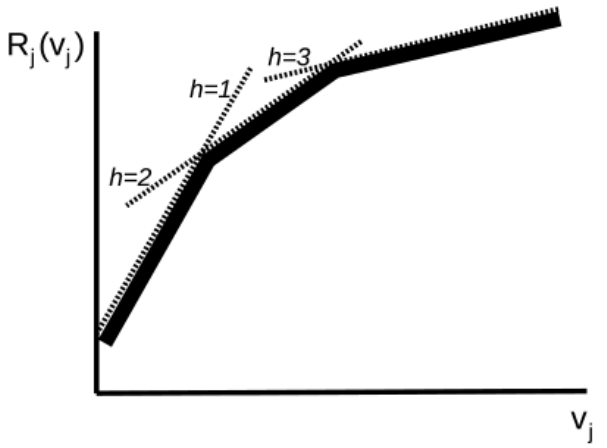

Figure 4.15 Unidimensional concave piecewise linear function represented by three hyperplanes.

Reservoirs dynamics constraints

The volume of water stored in reservoirs $j \in J$ evolves from a known initial condition $v_{j0}^{\omega} = V_j^{init}, \forall \omega \in \Omega$ (hm^3) according to discrete-time continuity equation

$$v_{jt}^{\omega} = v_{j,t-1}^{\omega} - Q_{jt}^{\omega,out} + Q_{jt}^{\omega,in} + \mathcal{I}_{jt}^{\omega}, \qquad \forall j, t, \omega \tag{4.13}$$

where v_{jt}^{ω} (hm^3) represents storage of reservoir j at the end of period t for scenario ω, $\mathcal{I}_{jt}^{\omega}$ is the natural inflows in reservoirs j during time periods t in scenario ω. The controlled outflow volume for reservoir j outlet is defined by

$$Q_{jt}^{\omega,out} = \sum_{c \in C} \sum_{i \in I(j)} (D_{ict}^{\omega} + q_{ict}^{\omega}) \beta \Delta t_c$$

where D_{ict}^{ω} (m^3 s^{-1}) and q_{ict} (m^3 s^{-1}) are decision variables representing the spillage and turbined outflow at plant i for load level c at time period t. The parameter $\beta = 0.0036$ is a constant for unit conversion. The controlled inflow for reservoir j is defined by

$$Q_{jt}^{\omega,in} = \sum_{u \in \mathcal{U}(j)} Q_{ut}^{\omega,out}$$

where the set $\mathcal{U}(j)$ contains reservoirs located immediately upstream of reservoir j.

Energy balance constraints

The constraints

$$\sum_{i \in I(z)} P_{ict}^{\omega} + \sum_{\ell \in L_+(z)} (1 - \epsilon_\ell) X_{\ell ct}^{\omega} - \sum_{\ell \in L_-(z)} X_{\ell ct}^{\omega} = d_{zct}^{net}, \quad \forall z, t, c, \omega \tag{4.14}$$

ensure that the net electrical load d_{zct}^{net} is satisfied for load level c, in zone z during time period t. Decision variables $X_{\ell ct}^{\omega}$ represents the power transmitted on link ℓ during time period t for load level c, P_{ict}^{ω} represents the power output of hydro plant i for load level c during time period t for scenario ω.

Hydroelectric generation constraints

The power output P_{ict}^{ω} of hydro plants $i \in I$ for load levels $c \in C$ at period t is returned by generation functions ϕ_i. Since these functions are concave and piecewise linear they can

be defined by taking the minimum value among different hyperplanes $h \in \mathcal{H}_i$ as follow:

$$\phi_i(q^{\omega}_{ict}, \tilde{v}^{\omega}_{j(i)\,t}, v^{\omega}_{\delta(i)\,t}) = \min_{h \in \mathcal{H}_i} \{\gamma^{hi}_0 + \gamma^{hi}_1 q^{\omega}_{ict} + \gamma^{hi}_2 \tilde{v}^{\omega}_{j(i)\,t} + \gamma^{hi}_3 v^{\omega}_{\delta(i)\,t}\}$$

where q^{ω}_{ict} is the turbined outflow, γ^{hi}_0, ..., γ^{hi}_3 are the coefficients of hyperplane h. The average storage during period t in the upstream reservoir $j(i)$ is

$$\tilde{v}^{\omega}_{j(i)\,t} = (v^{\omega}_{j(i)\,t} + v^{\omega}_{j(i)\,t-1})/2 \tag{4.15}$$

and $\tilde{v}^{\omega}_{\delta(i)\,t}$ is the average storage in reservoir $\delta(i)$ which is located immediately downstream of plant i is

$$\tilde{v}^{\omega}_{\delta(i)\,t} = (v^{\omega}_{\delta(i)\,t} + v^{\omega}_{\delta(i)\,t-1})/2. \tag{4.16}$$

In the model, this relationship is represented using the following constraints

$$P^{\omega}_{ict} \leq \gamma^{hit}_0 + \gamma^{hit}_1 q^{\omega}_{ict} + \gamma^{hit}_2 \tilde{v}^{\omega}_{j(i)\,t} + \gamma^{hit}_3 \tilde{v}^{\omega}_{\delta(i)\,t}, \qquad \forall h \in \mathcal{H}, i \in I, c \in C, t = 1, ..., T, \omega \in \Omega \tag{4.17}$$

The following inequality constraints

$$q^{\omega}_{ict} \leq \alpha^i_0 + \alpha^i_1 \tilde{v}_{j(i)\,t} + \alpha^i_2 \tilde{v}_{\delta(i)\,t}, \qquad \forall i \in I, c \in C, t = 1, ..., T, \omega \in \Omega \tag{4.18}$$

represents the maximum turbined outflow as a function of head. Linear coefficients α^i_0, α^i_1 and α^i_2 are known constants.

Box constraints

The following bounds

$$\underline{P}_{it} \leq P^{\omega}_{ict} \leq \overline{P}_{it}, \qquad \forall i, c, t, \omega \tag{4.19}$$

$$\underline{q}^{\min}_{it} \leq q^{\omega}_{ict} \leq \overline{q}_{it}, \qquad \forall i, c, t, \omega \tag{4.20}$$

$$\underline{D}^{\min}_{it} \leq D^{\omega}_{ict} \leq \overline{D}_{it}, \qquad \forall i, c, t, \omega \tag{4.21}$$

$$\underline{v}^{\min}_{jt} \leq v^{\omega}_{jt} \leq \overline{v}_{jt}, \qquad \forall j, t, \omega \tag{4.22}$$

$$\underline{X}_{\ell t} \leq X^{\omega}_{\ell ct} \leq \overline{X}_{\ell t}, \qquad \forall \ell, c, t, \omega \tag{4.23}$$

are imposed on all decision variables to represent physical limits of the power system or other operational constraints (e.g. minimum flow requirements).

Non-anticipativity constraints (NACs)

The following linear equality constraints

$$\begin{aligned} P^{\omega}_{ict(n)} - \hat{P}^{n}_{ic} &= 0, \quad \forall i, c, n, \omega \\ q^{\omega}_{ict(n)} - \hat{q}^{n}_{ic} &= 0, \quad \forall i, c, n, \omega \\ D^{\omega}_{ict(n)} - \hat{D}^{n}_{ic} &= 0, \quad \forall i, c, n, \omega \\ X^{\omega}_{\ell ct(n)} - \hat{X}^{n}_{\ell c} &= 0, \quad \forall \ell, c, n, \omega. \end{aligned}$$

ensure that all decision variables are scenario invariant at each tree node. Variables $\hat{P}^{n}_{ic}$, $\hat{q}^{n}_{ic}$ and $\hat{D}^{n}_{ic}$ represent the power output, turbined outflow and spilled outflow of hydro plant i for load level c at node n. Variables $\hat{X}^{n}_{\ell c}$ represent the power entering link ℓ for load level c at node n.

4.4.6 Implementation details

The PHA is implemented in object-oriented C++ using version 12.4 of ILOG CPLEX/Concert technology library. We use a personal computer running with a AMD Phenom II X6 2.8 GHz processor and 6 GB of RAM. The computer runs on the 64 bits version of Ubuntu 12.04.

4.5 Results

4.5.1 Sensitivity to the penalty parameter

The penalty parameter ρ weights quadratic penalties in $\mathcal{S}^{k}_{\omega}$ and corresponds to step size in dual vectors update formulas. This parameter plays a key role by controlling the rate at which feasibility and the objective value are improved. In practice, ρ should be chosen carefully to maximize the algorithm's performance.

Two different approaches are compared in this experiment. In the first approach, we use a penalty parameter which remains constant at each iteration. Theoretical results presented in Rockafellar et Wets (1991) are based on this approach. Different values of ρ are tested. In the second approach, the penalty parameter ρ_k is updated at each iteration $k = 0, 1, 2, ...$ using the classical update formula for general augmented Lagrangian methods Nocedal et Wright (2006)

$$\rho_{k+1} \leftarrow \mu \rho_k \tag{4.24}$$

where $\rho_0 > 0$ and $\mu > 1$ are known constant. Different values of ρ_0 and μ are tested. The aim of this experiment is to show the sensitivity of the PHA to the penalty parameter choice.

To perform this experiment in reasonable time, we only penalize violations of NACs at nodes $\mathcal{N}_* = \{0, ..., 170\}$. These nodes correspond to the first year of the planning horizon. We omitted NACs associated with tree nodes at the second year to ensure the computing time remains manageable.

Tables 4.5.1 and 4.5.1 summarizes the numerical results obtained with the first and second approaches, respectively. Updating the penalty parameter at each iteration requires more tuning, but this approach reduces substantially the algorithm's running time. Figure 4.16 shows the algorithm's progress at each iteration when a constant penalty parameter is used. Violation of NACs is large initially, but decreases rapidly in the first 10–12 iterations. Slower progress is observed afterwards. For large values of ρ, feasibility improves rapidly, but refinement of the objective value is slowed down. Conversely, a large number of iterations will be required to obtain a feasible solution if ρ is small. The larger this constant is, the lower is the number of iterations (and running time) and the lower is the objective value (maximization).

The algorithm converged after 28–102 iterations depending on which ρ_0 and μ were used. In general, using large values of ρ_0 and μ reduces the number of iteration, but lead to a lower objective value. In general, the running time is much lower when a variable penalty parameter is used in comparison with a constant one. The rate of progress obtained using a variable penalty parameter is shown in Figures 4.17–4.19. Feasibility improved rapidly when large values of ρ_0 were used, but converged to a lower objective value. The value of μ influenced importantly the rate at which feasibility improved. The higher is μ, the higher is the rate of improvement.

Table 4.3 Computational performance of the PHA using a constant penalty parameter. *Solution has not converged when $\rho = 10^{-6}$. We terminated the algorithm with $\zeta_k = 0.16$

ρ	Iterations	Time (hours)	Objective ($\times 10^9$\$)
10^{-4}	144	15.57	5.658657
10^{-5}	199	21.82	5.660786
10^{-6}	275*	29.74	5.660880

4.5.2 Water release at the first period

The volume of water to be released from each hydro plant at the first time period (week) is one of the most important model output. At Hydro-Québec, these outputs are generated at

Table 4.4 Computational performance of the PHA using a variable penalty parameter.

ρ_0	μ	Iterations	Time (hours)	Objective ($\times 10^9$\$)
10^{-7}	1.20	56	6.19	5.660827
10^{-7}	1.30	43	4.45	5.660778
10^{-7}	1.40	32	3.80	5.660736
10^{-6}	1.20	39	4.11	5.660448
10^{-6}	1.30	33	3.42	5.660018
10^{-6}	1.40	28	2.90	5.659469
10^{-5}	1.05	102	11.07	5.660825
10^{-5}	1.08	40	4.16	5.656266

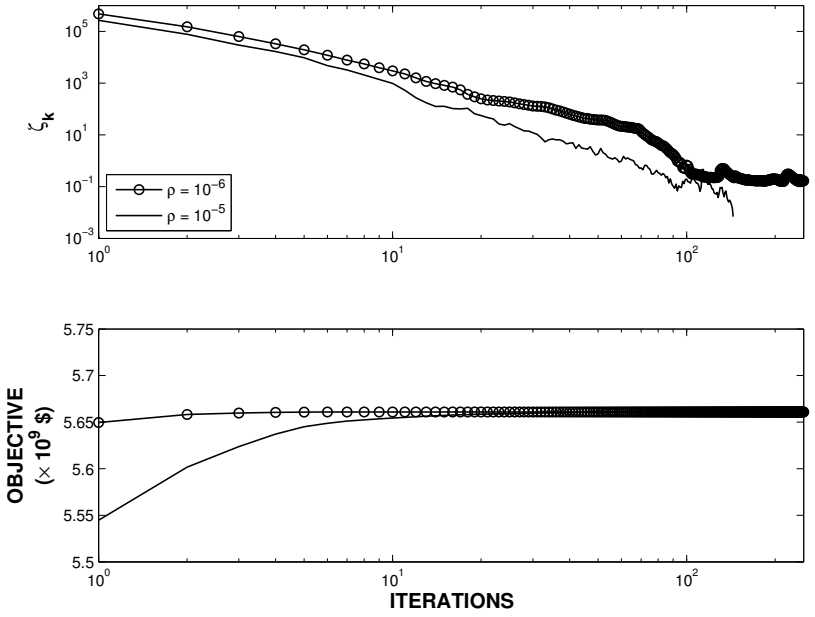

Figure 4.16 Convergence with a constant penalty parameter.

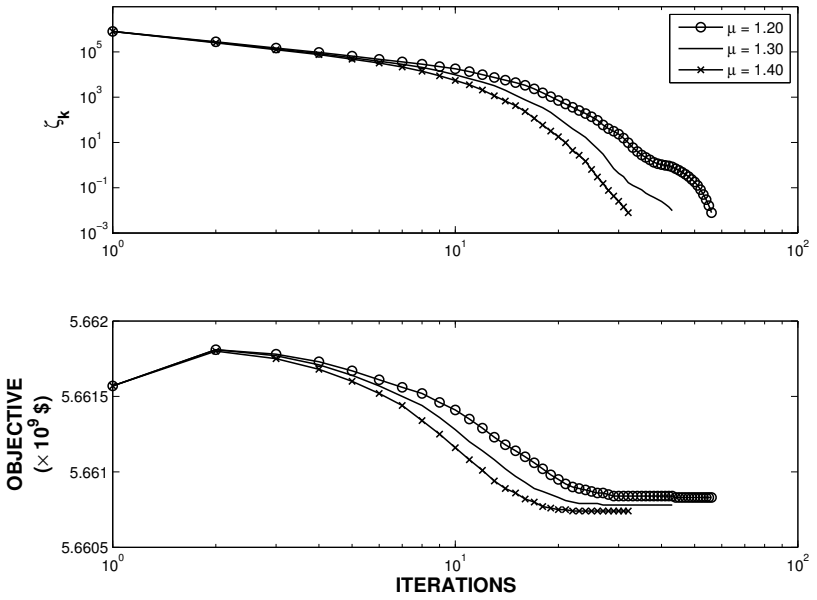

Figure 4.17 Convergence with a variable penalty parameter using $\rho_0 = 10^{-7}$.

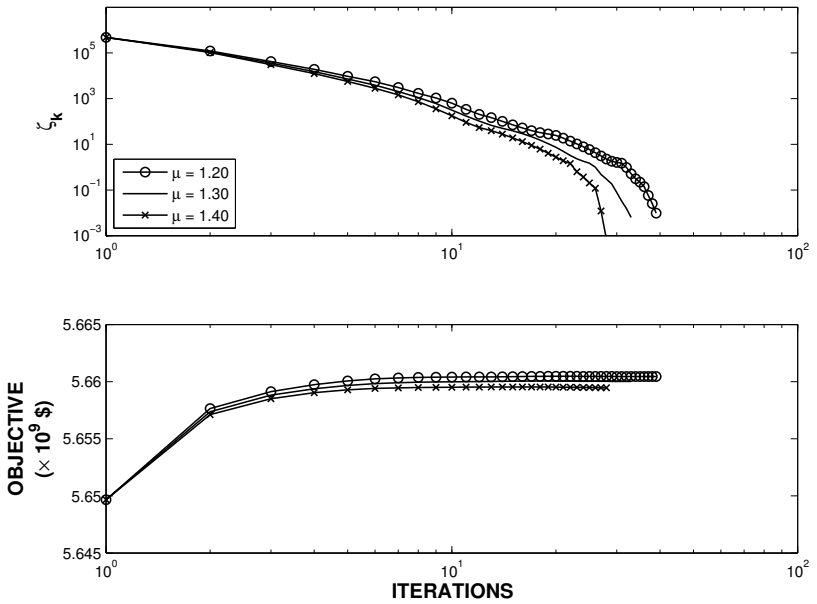

Figure 4.18 Convergence with a variable penalty parameter using $\rho_0 = 10^{-6}$.

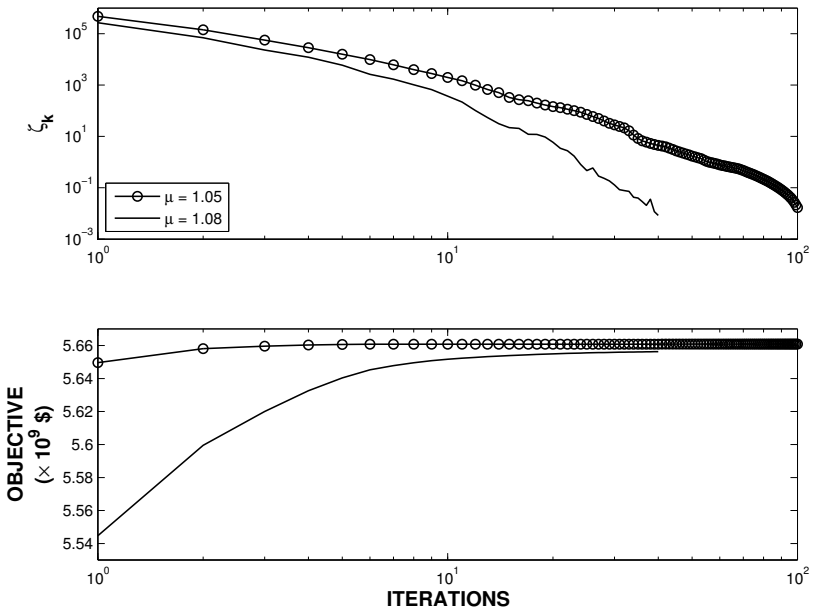

Figure 4.19 Convergence with a variable penalty parameter using $\rho_0 = 10^{-5}$.

the beginning of each week using newly available informations (e.g. inflows forecast, planned market transactions, ...) by system managers to give a weekly targets to the network's Independent System Operator (ISO). In this subsection, we compare the water release targets obtained by solving the stochastic and deterministic HQRMP. The stochastic solution is computed with the PHA using $\rho_0 = 10^{-7}$ and $\mu = 1.3$. All NACs are taken into account when solving the stochastic HQRMP. Consequently, the water release targets at the first period are scenario invariant. Before using the PHA, we attempted to compute an optimal stochastic solution by solving directly the deterministic equivalent stochastic HQRMP. Unfortunately, no optimal solution was found after 13.1 hours of running time. Deterministic water release targets are computed by solving the HQRMP sequentially for each inflow scenario in the tree. NACs were not taken into account when solving the deterministic HQRMP. Therefore, water release targets at the first period can vary depending on which scenario is considered. It is worth mentioning that the deterministic problem corresponds to a streamlined version of Hydro-Québec's.

Table 4.5.2 compare deterministic and stochastic solutions at the first time period. The symbols σ represents the standard deviation among the different scenarios. Deterministic solutions vary importantly depending on which input scenario is used in the model. This phenomenon reproduces what happens when the HQRMP is solved operationally by system managers for different scenarios. Choosing the right decision to apply in practice is not

obvious. Any given solution is only optimal for a particular scenario and can potentially have a very poor performance on another scenario. Each deterministic solution is optimal for its associated scenario. Furthermore, any particular solution can have a poor expected performance (low head, high spillage, infeasible) when used on another possible realization of the hydrological stochastic process. The scenario-averaged solution is potentially suboptimal generation efficiency and can even violate physical constraints. In the stochastic solution, the water release at the first period (root node $n = 0$) is scenario invariant for all hydro plants as shown on Tables 4.5.2. The stochastic solution returned by the PHA is optimal for a range of possible outcomes. Furthermore, physical constraints are satisfied.

The water release target is especially important for hydro plants

$$i \in I_{up} = \{1, 9, 10, 12, 14, 15, 20, 21, 24\}$$

since they are directly connected to the largest and most upstream reservoirs $j \in J_{up} = \{1, 8, 10, 11, 12, 16, 17, 20\}$ in the system. Most of the system's (unregulated) natural inflow volume arrives into reservoirs $j \in J_{up}$. In general, downstream reservoirs have a much smaller storage capacity and are mainly supplied by controlled inflow arriving from the upstream reservoirs. As shown on Tables 4.5.2, the water released from hydro plants $i \in I_{up}$ during $t = 1$ has a direct impact on future operating conditions (head, flexibility, ...) at downstream plants. The variability of deterministic solutions is large at most hydro plants. For example, the water release at plant $i = 1$ varies from 209.32 hm^3 to 679.03 hm^3 depending on which inflow scenario is considered. These different solutions would be useful only if the future inflow could be predicted perfectly, but this is obviously not the case. The stochastic optimization model returns a unique target of 553.17 hm^3 for all scenarios since it takes into account the fact that future inflow are uncertain. We can also observe that even the averaged deterministic solution of 560.70 hm^3 is significantly different from the stochastic solution at this plant. The same phenomenon is observed at most hydro plants in the system.

The (expected) objective value among the different deterministic solutions is 5.661925 $\times 10^9$\$. On average, it took 3.6 seconds to resolve each deterministic problem. The expected objective value returned by the PHA is 5.660100 $\times 10^9$\$. Convergence was attained after 42 iterations and took 8.1 hours of running time. On average, each scenario subproblem was solved in 11.5 seconds.

The difference between the objective value 1.8250 $\times 10^6$ \$ is relatively small in terms of percentage of the objective function. In introductory texts on stochastic programming

Table 4.5 Volume of water released from each hydro plant during $t = 1$ (hm^3).

	Deterministic				Stochastic	
i	Average	Maximum	Mininimum	σ	Average	σ
1	560.70	679.03	209.32	115.27	553.17	5.99E-02
2	555.75	667.05	270.86	96.88	541.85	9.44E-03
3	879.15	887.85	680.10	37.83	887.84	7.38E-03
4	1246.92	1254.46	1198.75	8.43	1248.27	6.51E-03
5	1359.18	1549.55	1352.62	34.54	1352.63	7.83E-03
6	2093.81	2191.93	2088.66	17.34	2090.77	4.57E-03
7	803.12	838.21	801.27	6.64	801.79	4.63E-03
8	2902.30	3034.38	2895.29	23.84	2897.92	8.68E-03
9	328.69	332.27	223.32	19.06	332.27	1.00E-03
10	421.78	425.80	307.15	20.42	425.80	1.16E-03
11	688.10	746.14	649.25	18.05	691.07	3.07E-03
12	1016.70	1056.24	1015.46	6.78	1015.46	3.13E-03
13	129.82	149.44	126.54	3.38	129.27	3.76E-03
14	316.73	392.16	297.33	20.60	311.39	2.11E-03
15	336.60	382.83	286.50	24.36	342.33	1.92E-03
16	668.79	764.50	633.40	27.29	669.18	3.35E-03
17	832.28	898.20	794.92	19.27	847.45	3.51E-03
18	271.95	278.12	262.13	2.07	288.60	1.72E-03
19	560.73	620.49	522.91	19.77	559.26	1.63E-03
20	151.89	166.68	121.99	19.66	166.67	1.66E-03
21	217.94	277.82	143.12	24.71	205.59	2.73E-03
22	220.04	279.92	145.21	24.71	207.68	2.73E-03
23	225.27	285.15	150.44	24.71	212.92	2.75E-03
24	245.57	252.42	245.25	1.20	245.36	1.14E-03
25	259.68	285.32	250.98	6.99	257.45	1.46E-02

(Birge et Louveaux, 2011), this difference is often defined as the the expected value of perfect information (EVPI) which can be used to quantify how important uncertainty is for a particular optimization problem. In reality, the inflow random vectors $\mathcal{I}_t$ are driven by a complex stochastic process characterized by continuous probability distribution functions. Because this process is represented coarsely using a discrete approximation of it, the EVPI computed in the experiment is only a biased estimator of the exact EVPI. Indeed, the EVPI computed in this experiment is only an underestimation of the real value. If more scenarios were included in the scenario tree, more NACs would have to be satisfied. Consequently, this would likely lead to a lower objective value (maximization) for the stochastic problem and a higher EVPI. Also, a relatively small EVPI would not imply that the problem is not a stochastic one. For example, Birge et Louveaux (2011) shows an example of problem whose EVPI is null and which possess a strictly positive value of stochastic solution (VSS). Computing a precise estimation of the VSS for the HQRMP would require a complex simulation experiment and is well beyond the scope of the current paper.

4.5.3 Reservoir state trajectories

In this subsection, we compare reservoir state trajectories obtained by solving the deterministic and stochastic HQRMP. Figures 4.20 and 4.25 show deterministic and stochastic trajectories for reservoir $j = 1$ which possess the largest storage capacity in the system, but possess only 469 MW of installed capacity. Figures 4.26 and 4.27 show deterministic and stochastic trajectories for reservoir $j = 4$ which possess 7,079 hm^3 of storage capacity and 2,779 MW of installed capacity. Head variations are critical at this reservoir. Figures 4.24–4.27 show box plots of weekly storage for reservoirs $j = 1$ and $j = 4$. Each box is represented by a thick line whose upper and lower limits represent the 25^{th} and 75^{th} percentiles, respectively. The central dot represents the median. The upper and lower limit of the thin line represent the most extreme values.

During the first winter season ($t = 1, ..., 12$), deterministic solutions differ importantly and the model exploits informations about future inflows that are not available to the decision maker. This is observed for both reservoirs $j = 1$ and $j = 4$. The targeted water level at the end of winter is highly scenario dependent in deterministic trajectories. In practice, spring flood timing and intensity is highly unpredictable and variable from year to year. At reservoirs where head variations are very important (e.g. at $j = 4$), the deterministic model returns particularly poor solutions. By assuming that the future spring flood is known with certainty, it typically empties these reservoirs just enough to fill to it to full capacity with the upcoming spring flood in order to maximize head during the summer season. The model

doesn't keep any safety margin in case that the spring flood would be different. In practice, if reservoir inflows are larger than the input scenario unproductive spillage will be generated. Conversely, if inflows are weaker than the input scenario, head will be lower during summer.

The stochastic trajectory during the first winter season is much easier to interpret since it is identical for all scenarios. For $t > 12$, the stochastic solution possess a tree-like structure. Branching in state trajectories correspond to branching nodes in the scenario tree. Furthermore, it satisfies all physical constraints of the power system. This is not necessarily the case for the average of deterministic solutions. Furthermore, the performance associated with the average of deterministic solutions can be suboptimal.

State trajectories for reservoir $j = 4$ are characterized by a well-defined annual drawdown-refill cycle. This reservoir is refilled at the beginning of each winter season (at t =1, 42 and 92). In practice, this is necessary to maximize head at hydro plant $i = 4$ (2,779 MW) to ensure that the power system possess enough available power during the peak load period. Conversely, trajectories for reservoir $j = 1$ do not have a clear annual cycle and have an upward trend. This is mainly due to the fact that reservoirs $j = 1$ and $j = 4$ have a quite different initial condition. Initially, reservoir $j = 4$ is filled at 100% of its capacity and reservoir $j = 1$ is only at 22% of its maximal capacity. This type of initial conditions are quite typical during this part of the year for the Hydro-Québec reservoir system. In fact, reservoir $j = 1$ is mainly used for potential energy storage since it is located upstream from other reservoirs in the same subsystem. Approximately half of the system's installed capacity is located downstream of reservoir $j = 1$.

4.5.4 Sensitivity to the anticipativity level

Deterministic solutions are fully anticipative since they are based on the assumption that future hydrological conditions are known with great precision over the entire planning horizon. At the opposite, the fully non-anticipative stochastic solution is obtained by satisfying NACs associated with every tree node $n \in \mathcal{N}_*$. In this experiment, we also compute partially-anticipative stochastic solutions are obtained by considering NACs associated with only a subset of tree nodes $\tilde{\mathcal{N}} \subset \mathcal{N}_*$. Tree nodes $n \in \tilde{\mathcal{N}}$ at which NACs are considered correspond to time periods $t = 1, 2, ..., \tilde{T}$. Table 4.6 summarizes all cases that were considered.

Table 4.7 shows numerical results obtained for cases A to F. The total running time and the time per subproblem increase rapidly from case A to case F. Penalizing violations of NACs increased the amount of average running time per subproblem.

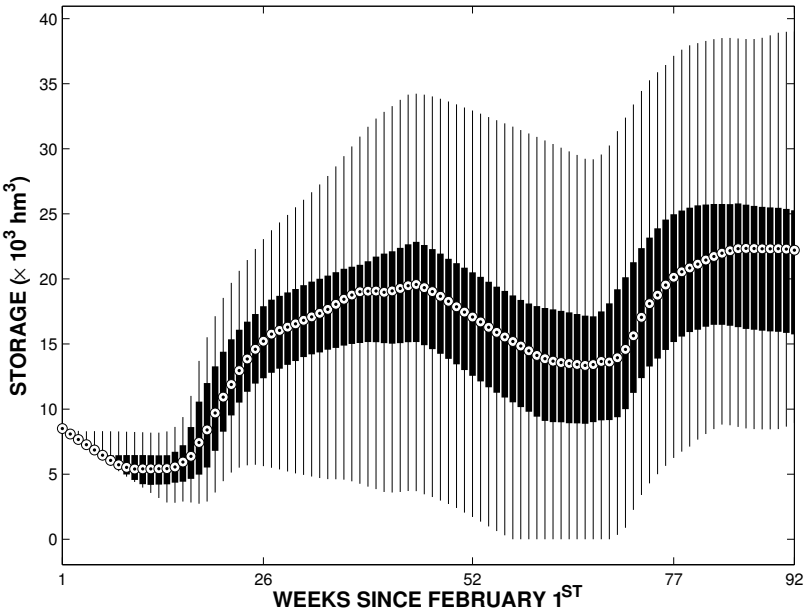

Figure 4.20 Deterministic state trajectories for reservoir $j = 1$.

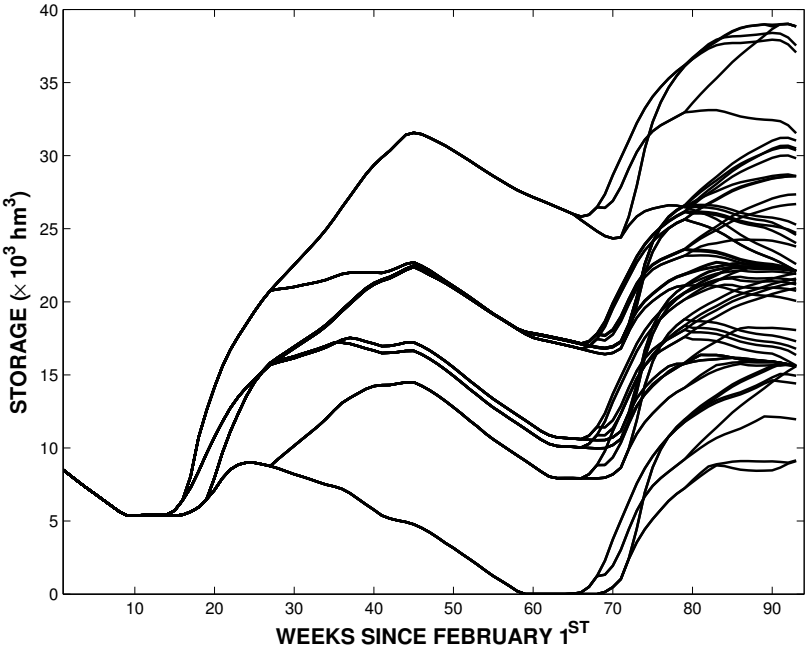

Figure 4.21 Stochastic state trajectories for reservoir $j = 1$.

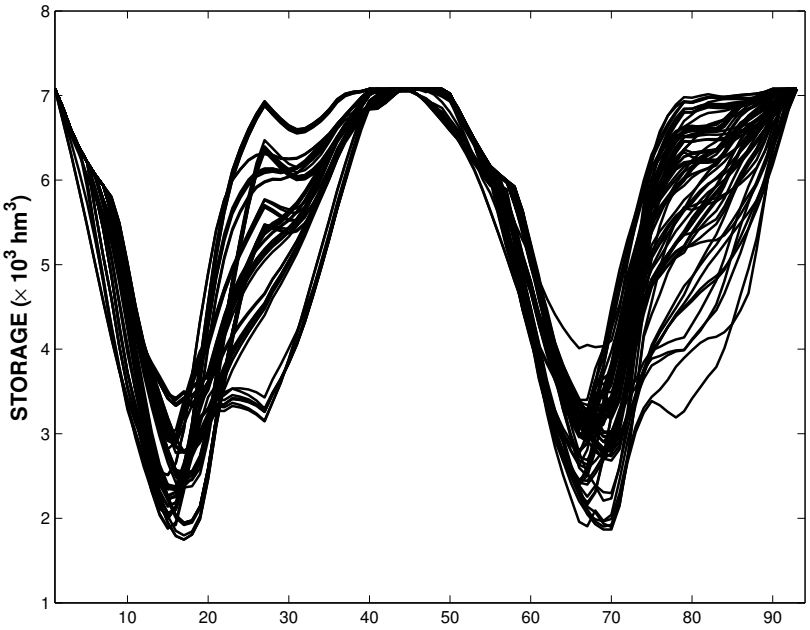

Figure 4.22 Deterministic state trajectories for reservoir $j = 4$.

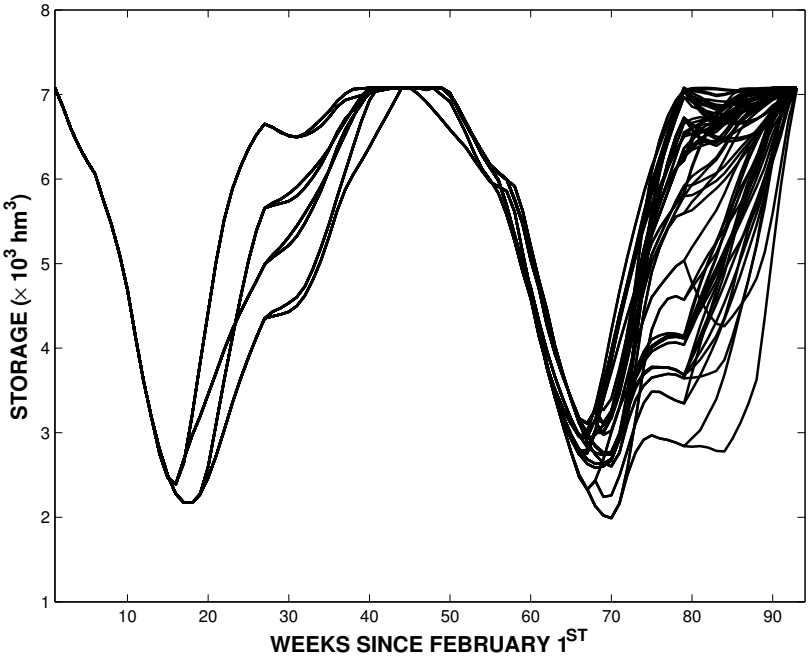

Figure 4.23 Stochastic state trajectories for reservoir $j = 4$.

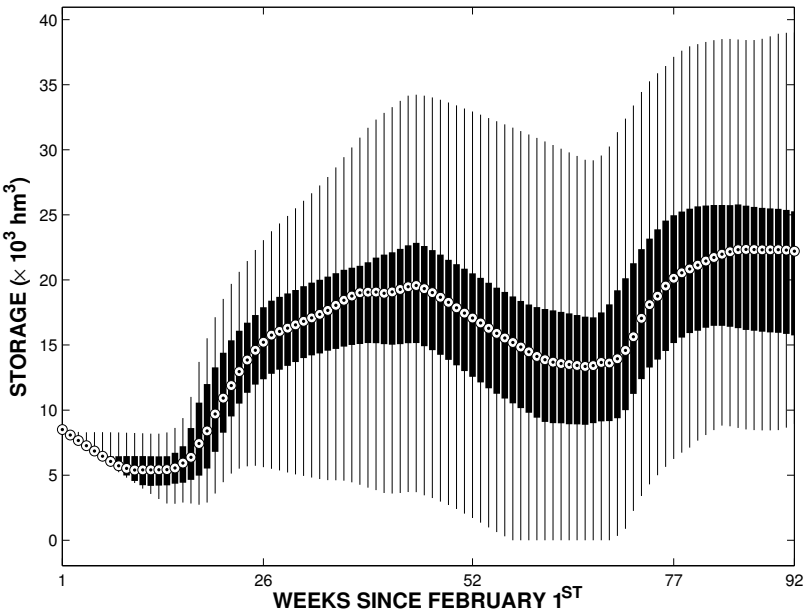

Figure 4.24 Box plots of deterministic state trajectories for reservoir $j = 1$.

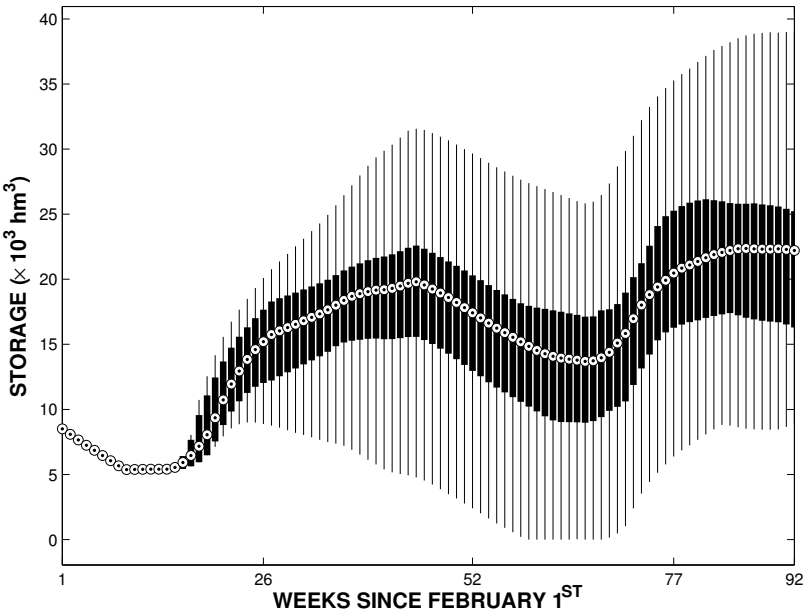

Figure 4.25 Box plots of stochastic state trajectories for reservoir $j = 1$.

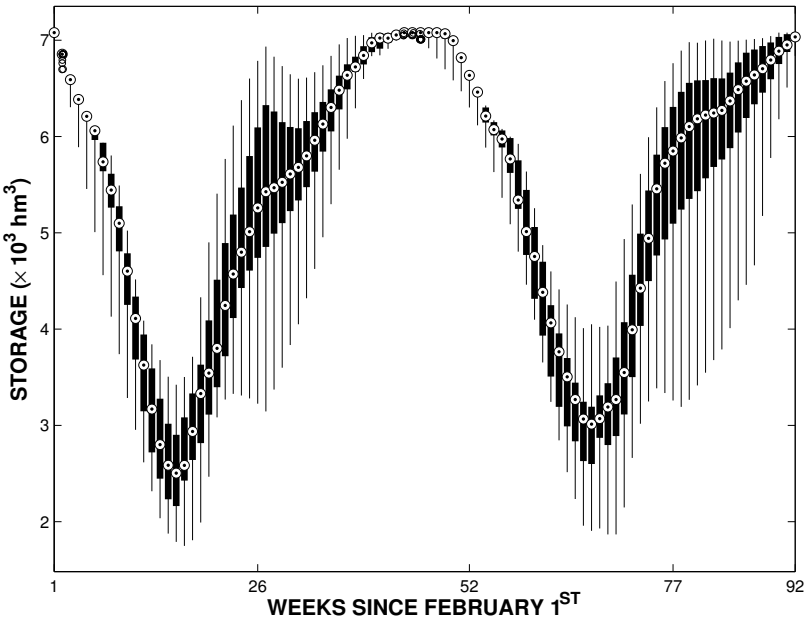

Figure 4.26 Box plots of deterministic state trajectories for reservoir $j = 4$.

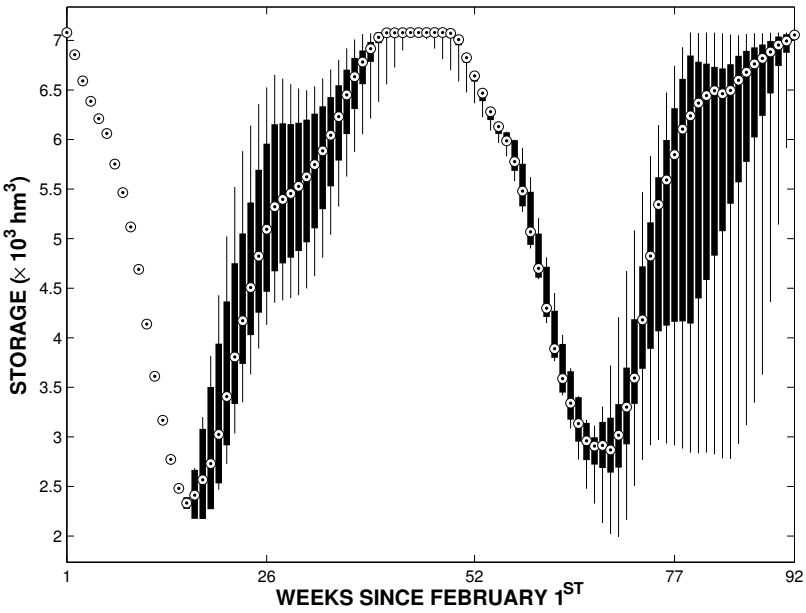

Figure 4.27 Box plots of stochastic state trajectories for reservoir $j = 4$.

Table 4.6 Cases considered.

Case	Anticipativity level	$\tilde{T}$	$\tilde{\mathcal{N}}_*$	Number of penalty terms
A	Fully anticipative (Deterministic)	0	0	486
B	Partially anticipative	12	12	5,832
C	Partially anticipative	40	171	19,440
D	Partially anticipative	64	363	31,104
E	Fully non-anticipative (Stochastic)	78	739	37,908

Table 4.7 Numerical results for cases A to E.

	Total time (minutes)	Time per subproblem (seconds)	Objective ($\times 10^9$ \$)
A	3.6	3.6	5.661925
B	105.9	4.6	5.661585
C	278.8	6.5	5.660778
D	380.7	9.1	5.660466
E	484.7	11.5	5.660100

4.6 Conclusions

In this paper, we demonstrate the applicability of the PHA for solving long-term RMPs. This method can be used to build a new stochastic optimization model upon an existing deterministic one which can be used to solve scenario subproblems. Only linear-quadratic penalty terms must be included in the objective function to penalize violations of NACs. We apply this approach on Hydro-Québec's reservoir system with stochastic inflows. We consider the most difficult problem facing the company's system managers: spring flood management. Any other hydro-climatic conditions facing system managers would be easier to deal with.

We test the PHA's performance for different penalty parameter values. Our results show the importance of using a variable penalty parameter rather than using a constant one. We also compare the stochastic solution obtained using the PHA with deterministic solutions. Deterministic solutions are quite sensitive to which inflow scenario is considered and are difficult to interpret. The stochastic solution are scenario-invariant and more robust that deterministic ones. Finally, we compute partially anticipative solutions by penalizing violations of NACs only at a subset of tree nodes. Numerical results show that NACs in the latter part of the horizon have an impact on the first-stage decisions.

The results obtained in this study are showing that the PHA is a promising method which could eventually be implemented at Hydro-Québec. Further research is needed for the

construction of the scenario tree to describe hydrologic uncertainty. It is worth mentioning that the approach use in this paper could easily be adapted to deal with other contexts. For example, thermal plants and market transactions could be integrated in the optimization models. Additional sources of uncertainty could also be included in the scenario tree. For example, the power and water demand, market prices or wind generation could eventually be treated as random parameters.

CHAPITRE 5

ARTICLE 2 : OPTIMAL SCENARIO SET PARTITIONING FOR MULTISTAGE STOCHASTIC PROGRAMMING WITH THE PROGRESSIVE HEDGING ALGORITHM

Pierre-Luc Carpentier[1,2], Michel Gendreau[1,2] and Fabian Bastin[1,3]

[1]Centre interuniversitaire de recherche sur les réseaux d'entreprise, la logistique et le transport (CIRRELT), Montréal, Québec, Canada.

[2]Department of Mathematics and Industrial Engineering, École Polytechnique de Montréal, Montrál, Québec, Canada.

[3]Department of Computer Science and Operations Research, University of Montreal, Montréal, Québec, Canada.

Soumis le 2 octobre 2013 au European Journal of Operational Research

Abstract

In this paper, we propose a new approach to reduce the total running time (RT) of the progressive hedging algorithm (PHA) for solving multistage stochastic programs (MSPs) defined on a scenario tree. Instead of using the conventional scenario decomposition scheme, we apply a multi-scenario decomposition scheme and partition the scenario set in order to minimize the number of non-anticipativity constraints (NACs) on which an augmented Lagrangian relaxation (ALR) must be applied. Our partitioning method is a heuristic algorithm that takes into account the complex branching structure of general multistage scenario trees. Minimizing the number of relaxed NACs (RNACs) enhances the PHA's convergence rate by decreasing the variability of subproblems solutions at duplicated tree nodes. This is due to the fact that minimizing the RNACs reduces the anticipativity level of subproblems by increasing the branching level of subtrees. This makes RNACs easier to satisfy. Our partitioning method also reduces the total RT per iteration in two different ways. Firstly, it decreases the number of linear and quadratic penalty terms that need to be included in subproblem's objective functions. Secondly, it reduces the total number of duplicated decision variables and

constraints at tree nodes that are associated with RNACs. The proposed method is tested on an hydroelectricity generation scheduling problem covering a 52-week planning horizon.

5.1 Introduction

Practical applications of stochastic programming (SP) methods for solving uncertain optimization problems are quite numerous and cover a wide spectrum of domains (Dupačová, 2002; Rusczynzki et Shapiro, 2003). The book by Gassman et Ziemba (2013) presents different applications of these methods in energy, logistics and production planning, finance and telecommunications. Most of these applications contain one or several random parameters (e.g. inflows, prices, interest rates, yield, demand, electrical load, wind/solar generation) that are characterized by a (joint) continuous probability distribution. A popular approach to represent continuously distributed parameters in SP models consists in replacing the original continuous distribution by a discrete distribution possessing a finite number of possible outcomes (scenarios). This type of approximation leads to a scenario tree representation of uncertainty. Fig. 5.1a shows a simple example of a scenario tree with three stages, four scenarios and seven nodes. Different methods were proposed over the years for constructing a scenario tree from a set of synthetic or historical time series (e.g. Pflug, 2001; Høyland et Wallace, 2001; Latorre *et al.*, 2007). Heitsch et Römisch (2009) proposed a construction method based the theoretical results of Heitsch *et al.* (2006). The SCENRED2 package of the General Algebraic Modeling System (GAMS) is a computer implementation of this technique.

Multistage stochastic programs (MSPs) defined on scenario trees can be reformulated into deterministic equivalent programs (DEPs) of finite size (variables, constraints). The number of decision variables and constraints is usually proportional to the number of nodes contained in the scenario tree. Therefore, the size of DEPs grows exponentially with the discretization level (number of branching stages, number of branches per stage) used to describe the original probability distribution. In general, most real-world MSPs cannot be solved directly using a commercial solver (e.g. GLPK, Gurobi, XPress-MP, CPLEX) when an accurate representation of the random parameter is used. The required amount of random access memory (RAM) is typically the main limiting factor when solving large-scale linear or quadratic programs directly.

Over the past decades, different decomposition methods were proposed in the literature for solving efficiently large-scale SPs (Rusczynzki, 2003; Sagastizabal, 2012). Solution methods based on Benders (1962) decomposition (e.g. Van Slyke et Wets, 1969; Birge, 1985; Birge

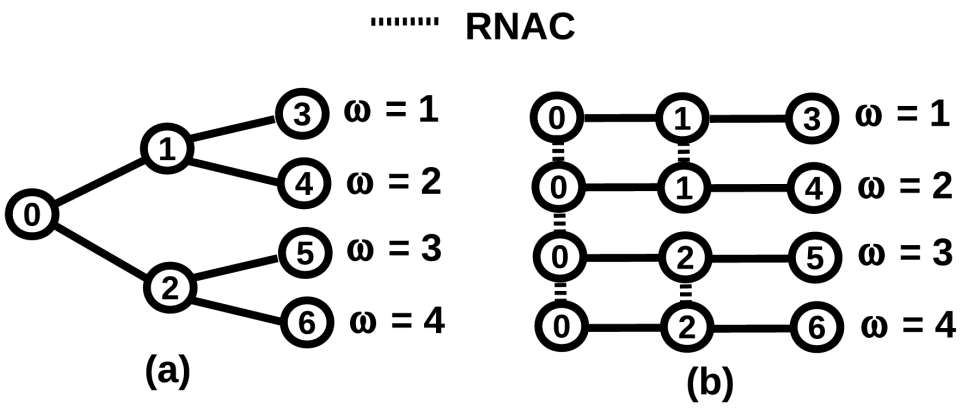

Figure 5.1 (a) Example of a scenario tree. (b) Illustration of the scenario-decomposition scheme.

et Louveaux, 1988; Pereira et Pinto, 1991; Laporte et Louveaux, 1993; Küchler et Vigerske, 2007; Carpentier *et al.*, 2013a) are widely used for solving linear MSPs. These methods use a stage-wise decomposition scheme and work by constructing convex and piecewise linear recourse functions. The progressive hedging algorithm (PHA) proposed by Rockafellar et Wets (1991) is another popular method for solving SPs defined on scenario trees. This method was applied successfully in various type of problems including electricity generation planning (Goncalves *et al.*, 2011; Carpentier *et al.*, 2013b), network design (Crainic *et al.*, 2011), network flow (Mulvey et Vladimirou, 1991), lot-sizing (Haugen *et al.*, 2001) and resources allocation (Watson et Woodruff, 2011). To use the traditional version of the PHA, a decision vector $x_{t\omega} \in \mathbb{R}^m$ must be defined at each stage (time period) $t = 1, ..., T$ for all scenarios $\omega \in \Omega$ contained in the tree. All non-anticipativity constraints (NACs) must formulated explicitly as linear equality constraints

$$x_{t(n)\omega} = \hat{x}_n, \qquad \forall \omega \in \Omega, n \in \mathcal{N}^*_\omega : \lambda_{n\omega} \tag{5.1}$$

to ensure that feasible solutions are scenario-invariant at each tree node. The function $t(n)$ returns the stage index associated with tree node n, $\hat{x}_n \in \mathbb{R}^m$ is the decision vector at node n, $\lambda_{n\omega} \in \mathbb{R}^m$ is the vector of Lagrange multipliers associated with NACs for scenario ω at node n, $\mathcal{N}^*_\omega$ is a set of duplicated nodes that are visited by scenario ω. By definition, duplicated nodes are the ones that are visited by two scenarios or more. The traditional version of the PHA works by applying an augmented Lagrangian relaxation (ALR) on all constraints (5.1). Then, a scenario-decomposition scheme is applied on the resulting optimization problem. In practice, the running time of the PHA can become too long to be applicable if the convergence

rate is too slow or if the running time per iteration is too important. This can happen if the number of NACs is large or if scenarios differ by much. Indeed, the number of linear-quadratic penalty terms depends directly on the number of NACs to be satisfied. The required number of iterations to converge is sensitive to the variability among the subproblem's solutions at tree nodes that are visited by two scenarios or more. The performance of the PHA is often reported to be sensitive to the penalty parameter choice. Unfortunately, tuning this parameter can be a time consuming task. Fig. 5.1b illustrates an application of the scenario-decomposition scheme on the scenario tree shown on Fig. 5.1a. In this example, $\mathcal{N}_1^* = \mathcal{N}_2^* = \{0, 1\}$ and $\mathcal{N}_3^* = \mathcal{N}_4^* = \{0, 2\}$. An ALR must be applied on the following NACs

$$x_{1,1} = x_{1,2} = x_{1,3} = x_{1,4} = \hat{x}_0, \tag{5.2}$$

$$x_{2,1} = x_{2,2} = \hat{x}_1,\ x_{2,3} = x_{2,4} = \hat{x}_2. \tag{5.3}$$

A different version of the PHA was proposed by Crainic *et al.* (2013) for solving two-stage stochastic network design problems. Instead of applying the traditional scenario-decomposition scheme, these authors applied a multi-scenario decomposition scheme. With their approach, each subproblem is a small two-stage SP (TSP) defined on a group of scenarios. These authors partition the scenario set Ω into disjoint scenario groups Ω_c for $c \in \mathcal{C}$. Scenarios are grouped according to their similarity or dissimilarity level. The results of this study show that using such a partitioning method enhanced the performance of the PHA for solving TSPs. However, their approach is designed specifically for solving TSPs and it could not be used directly when dealing with MSPs. Indeed, the similarity-based method proposed by Crainic *et al.* (2013) works well when TSPs are considered because the topology of two-stage scenario tree is quite simple. In any two-stage scenario trees, all leaves share the same ancestor node (the root) and, consequently, the total number of RNACs is always equal to the number of scenario groups. This is true no matter which scenarios are grouped together. Unfortunately, this is usually not the case for multistage scenario trees (MSTs). Indeed, MSTs usually possess a complex branching structure and, because of this, the resulting number of RNACs will depend on which scenarios are grouped together. Fig. 5.2 shows a simple illustrative example where two different partitioning schemes are applied to the tree shown on Fig. 5.1a. For the scheme shown on Fig. 5.2a, the groups $c = 1$ and $c = 2$ are defined on scenarios $\Omega_1 = \{1, 2\}$ and $\Omega_2 = \{3, 4\}$, respectively. With this scheme, only the two following NACs need to be formulated explicitly (and relaxed)

$$\tilde{x}_{0,1} = \tilde{x}_{0,2} = \hat{x}_0$$

where $\tilde{x}_{nc}$ is the decision at node n in group c. The remaining NACs associated with nodes 1

and 2 are dealt with directly when solving subproblems defined on groups 1 and 2, respectively. For the scheme shown on Fig. 5.2b, the groups $c = 1$ and $c = 2$ are defined on scenarios $\Omega_1 = \{1, 3\}$ and $\Omega_2 = \{2, 4\}$, respectively. The following NACs need to be formulated explicitly

$$\tilde{x}_{0,1} = \tilde{x}_{0,2} = \hat{x}_0,\ \tilde{x}_{1,1} = \tilde{x}_{1,2} = \hat{x}_1,\ \tilde{x}_{2,1} = \tilde{x}_{2,2} = \hat{x}_2.$$

The aim of this paper is to propose a new scenario set partitioning method designed specifically for solving MSPs. The proposed partitioning method is a heuristic algorithm designed to build an optimal partition of Ω that minimizes the total number of relaxed NACs (RNACs). Our method takes into account the complex branching structure of general multistage scenario trees. In the context of MSPs, minimizing the total number of relaxed NACs (RNACs) enables to reduce the running time of the PHA for many different reasons.

- Firstly, it makes subproblems easier to solve by decreasing the number of linear and quadratic terms that need to be added in subproblem's objective functions to penalize violations of RNACs. This effect reduces the running time per iteration.
- Secondly, it reduces the total number of constraints and decision variables that need to be duplicated in all subproblems. In turn, this effect can also reduce the running time per iteration.
- Thirdly, minimizing the number of RNACs decreases the variability among subproblems solutions at duplicated tree nodes $n \in \mathcal{N}_c^*$ by enhancing the branching level in scenario subtrees. This effect accelerates the PHA's convergence rate by making RNACs easier to satisfy.

The paper is organized as follows. A general mathematical formulation for MSPs is presented in Section 5.2. Section 5.3 presents a non conventional version of the PHA which is based on a multi-scenario decomposition scheme. Section 5.4 describes the optimal scenario set partitioning method. Section 5.5 describes a numerical experiment. Numerical results are presented in Section 5.6. Comments and conclusions are drawn in Section 5.7.

5.2 Problem formulation

5.2.1 Multistage stochastic program

We consider the following optimization problem defined on T time periods

$$(\mathcal{P}) \qquad \min_{x_t} \mathbb{E}\left[\sum_{t=1}^{T} g_t(x_t, \xi_t)\right] \tag{5.4}$$

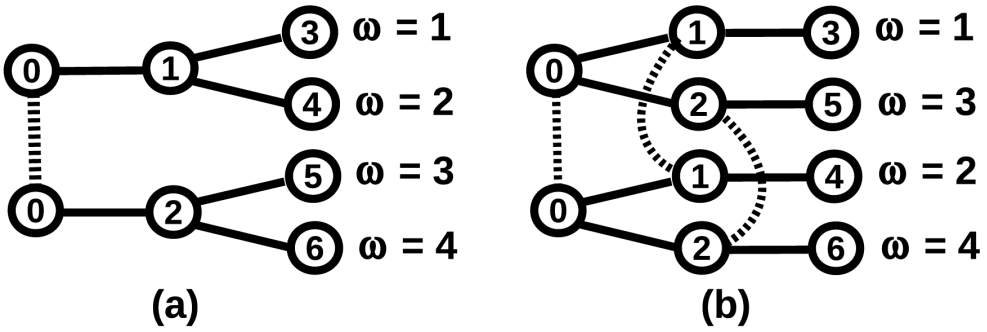

Figure 5.2 Examples of grouping schemes for a MST.

subject to

$$A_t(\xi_t)x_t + B_t(\xi_t)x_{t-1} = b_t(\xi_t) \quad , \quad \forall t = 1, ..., T \tag{5.5}$$

$$x_t \in \mathcal{X}_t \quad , \quad \forall t = 1, ..., T \tag{5.6}$$

$$x_t \in \mathcal{F}_t(\xi_1, ..., \xi_t) \quad , \quad \forall t = 1, ..., T \tag{5.7}$$

where $\mathbb{E}[\,\cdot\,]$ is the expectation operator, $g_t(\cdot, \cdot)$ are convex functions representing the operating costs at time period t, x_t is the decision vector at time period t and ξ_t is a random vector at time period t. Each component of ξ_t represents one of the problem's random parameter (e.g. interest rate, price, demand, inflows, ...) during time period t. Equations (5.5) and (5.6) represent dynamic (e.g. water budget, inventory dynamics, ramping constraints, ...) and static (e.g. system limits, mechanisms, ...) constraints, respectively. Coefficient of technological matrices A_t, B_t and vectors b_t are treated as random variables. The sets of static constraints $\mathcal{X}_t$ are assumed to be non-empty and convex. Non-anticipativity constraints (5.7) ensure that each x_t is chosen using only known informations $\xi_1, ..., \xi_t$ and cannot depend on future (unknown) realizations of random parameters $\xi_{t+1}, ..., \xi_T$. Each set $\mathcal{F}_t$ contains all solutions that meet this criterion at time t.

5.2.2 Scenario tree

Each random vector ξ_t is characterized by a known probability distribution $\mathbb{P}_t$ which is conditional to previous observations $\xi_{t-1}, ..., \xi_{t-p}$ made over the $p \geq 1$ last periods. We assume that $\mathbb{P}_t$ possesses a finite number of possible outcomes at each $t = 1, ..., T$. We also assume that all random parameters are exogenous to the controlled system. Therefore, $\mathbb{P}_t$ is not influenced by x_t. Making these assumptions enables us to represent the stochastic

process $\{\xi_t : t = 1, ..., T\}$ using a finite scenario tree $\mathcal{T}$. Each node $n \in \mathcal{N}$ has an occurrence probability of p_n. Each scenario $\omega \in \Omega$ in $\mathcal{T}$ corresponds to a path from the root $0 \in \mathcal{N}$ to a particular leaf $\ell(\omega) \in \mathcal{L}$. The probability of a given scenario ω corresponds to the probability $p_{\ell(\omega)}$ of its leaf $\ell(\omega)$. The branching structure of $\mathcal{T}$ is represented by a function $a(n)$ which returns the ancestor of any node n. The vector ξ_n represents the realization of random parameters at node n. The set $\Omega(n)$ contains all scenarios visiting node n. Each set $\Delta(n)$ contains all child nodes of n. Fig. 5.1a shows a simple example with $T = 3$, $\mathcal{N} = \{0, 1, 2, 3, 4, 5, 6\}$, $\mathcal{L} = \{3, 4, 5, 6\}$, $\Omega = \{1, 2, 3, 4\}$, $a(1) = a(2) = 0$.

5.2.3 Deterministic equivalent program

The problem $\mathcal{P}$ defined on a scenario tree $\mathcal{T}$ can be transformed into the following DEP

$$(\mathcal{E}) \qquad \min \sum_{n \in \mathcal{N}} p_n g_{t(n)}(\hat{x}_n, \xi_n) \tag{5.8}$$

subject to

$$A_n \hat{x}_n + B_n \hat{x}_{a(n)} = b_n \quad , \quad \forall n \in \mathcal{N}, \tag{5.9}$$

$$\hat{x}_n \in \mathcal{X}_n \quad , \quad \forall n \in \mathcal{N}. \tag{5.10}$$

The program $\mathcal{E}$ is obtained from $\mathcal{P}$ by replacing all occurrences of stage-wise decision vectors x_t by a node-wise decision vector $\hat{x}_n$ at node n. In the objective function, the expectation operator is replaced by a finite sum. Each term is weighted by the probability of the corresponding tree node. Constraints (5.9) and (5.10) correspond to constraints (5.5) and (5.6), respectively. Non-anticipativity constraints (5.7) of $\mathcal{P}$ are represented implicitly in $\mathcal{E}$ because of the node-wise index system that we use. Indeed, any feasible solution to $\mathcal{E}$ is scenario-invariant at all tree nodes $n \in \mathcal{N}$. Therefore, x_n only depends on available informations at time period $t(n)$.

5.3 Solution method

5.3.1 Decomposition scheme

Instead of using the single-scenario decomposition scheme that is conventionally used with the PHA, we apply a multi-scenario decomposition scheme on $\mathcal{E}$. The scenario set Ω is partitioned into disjoint subsets (groups) Ω_c where $c \in \mathcal{C}$ is the group index. In this paper, we denote by $\mathcal{T}_c$ the subtree associated with all scenarios $\omega \in \Omega_c$ in group c. Each set $\mathcal{N}_c$ contains all the nodes that are visited by the scenarios $\omega \in \Omega_c$ in group c. The occurrence

probability of group c is defined as follow

$$\tilde{p}_c := \sum_{\omega \in \Omega_c} p_{\ell(\omega)}. \tag{5.11}$$

The probability $\hat{p}_{nc}$ of node n conditional to group c is defined as follow

$$\hat{p}_{nc} := p_n / \tilde{p}_c. \tag{5.12}$$

The conventional single-scenario decomposition scheme is a particular case of the multi-scenario decomposition scheme where $|\Omega_c| = 1$ for all $c \in \mathcal{C}$.

5.3.2 Mathematical formulation

The program $\mathcal{E}$ defined on scenario groups $c \in \mathcal{C}$ is reformulated into the following equivalent program

$$(\tilde{\mathcal{E}}) \qquad \min \sum_{c \in \mathcal{C}} \tilde{p}_c \sum_{n \in \mathcal{N}_c} \hat{p}_{nc} g_{t(n)}(\tilde{x}_{nc}, \xi_n) \tag{5.13}$$

subject to

$$A_n \tilde{x}_{nc} + B_n \tilde{x}_{a(n)c} = b_n \quad , \quad \forall c \in \mathcal{C},\ n \in \mathcal{N}_c \tag{5.14}$$

$$\tilde{x}_{nc} \in \mathcal{X}_n \quad , \quad \forall c \in \mathcal{C},\ n \in \mathcal{N}_c, \tag{5.15}$$

$$\tilde{x}_{nc} = \hat{x}_n \quad , \quad \forall c \in \mathcal{C},\ n \in \mathcal{N}_c^* \ :\ \tilde{\lambda}_{nc}. \tag{5.16}$$

The formulation $\tilde{\mathcal{E}}$ can be obtained from $\mathcal{E}$ by making the following transformations. The objective function (5.13) and constraints (5.14)–(5.15) are obtained by replacing all occurrences of $\hat{x}_n$ at nodes $n \in \mathcal{N}$ by an alternative decision vector $\tilde{x}_{nc}$ defined at nodes $n \in \mathcal{N}_c$ visited by groups $c \in \mathcal{C}$. In (5.13), the probability p_n of node n is replaced by $\tilde{p}_c \hat{p}_{nc}$ according to (5.12). The NACs (5.16) ensure that any feasible solution of $\tilde{\mathcal{E}}$ is group-invariant at all tree nodes and $\tilde{\lambda}_{nc}$ is the vector of Lagrange multipliers associated with node n and group c. Each set $\mathcal{N}_c^*$ contains all nodes visited by group c and by at least another group. For the example shown on Fig. 5.2a, $\mathcal{N}_1^* = \mathcal{N}_2^* = \{0\}$.

5.3.3 Augmented Lagrangian

We apply an ALR on constraints (5.16) and the resulting objective function to be minimized is

$$\mathcal{A}_\rho(\tilde{x}, \hat{x}, \tilde{\lambda}) = \sum_{c\in\mathcal{C}} \tilde{p}_c \left(\sum_{n\in\mathcal{N}_c} \hat{p}_{nc} g_{t(n)}(\tilde{x}_{nc}, \xi_n) + \sum_{n\in\mathcal{N}_c^*} \left(\tilde{\lambda}'_{nc}(\tilde{x}_{nc} - \hat{x}_n) + \frac{\rho}{2}\|\tilde{x}_{nc} - \hat{x}_n\|^2 \right) \right)$$

subject to constraints (5.14)–(5.15). The penalty parameter ρ is a positive constant that needs to be tuned, $\tilde{x} = (\tilde{x}_{nc})$ is the vector of group-wise decision vectors, $\hat{x} = (\hat{x}_n)$ is the vector of node-wise decision vectors, $\tilde{\lambda} = (\tilde{\lambda}_{nc})$ is the vector of Lagrange multipliers associated with constraints (5.16). All vectors are assumed to be column vectors, $(\cdot)'$ is the transpose operator and $\|\cdot\|$ is the Euclidean norm.

5.3.4 Progressive hedging algorithm

The algorithm begins with an initial penalty parameter ρ_0, a suboptimal node-wise solution $\hat{x}^0 = (\hat{x}^0_n)$ and Lagrange multiplier $\tilde{\lambda}^0 = (\tilde{\lambda}^0_{nc})$. Then, at each iteration $k = 0, 1, 2, \ldots$, the two following steps are performed.

Step 1. Find a new group-wise solution $\tilde{x}^{k+1} = (\tilde{x}^{k+1}_{nc})$ by minimizing $\mathcal{A}_{\rho_k}(\tilde{x}, \hat{x}^k, \tilde{\lambda}^k)$ for $\tilde{x}$ subject to constraints (5.14)–(5.15). Because we consider $(\tilde{x}^k, \tilde{\lambda}^k)$ as fixed values, this problem is separable by group. Each group-subproblem is a relatively small MSP defined on a subtree $\mathcal{T}_c$ associated with a particular group c. The subproblem associated with group c is

$$(\mathcal{S}^k_c) \qquad \min \tilde{p}_c \sum_{n\in\mathcal{N}_c} \hat{p}_{nc} g_{t(n)}(\tilde{x}_{nc}, \xi_n) + \sum_{n\in\mathcal{N}_c^*} \left((\tilde{\lambda}^k_{nc})'(\tilde{x}_{nc} - \hat{x}^k_n) + \frac{\rho_k}{2}\|\tilde{x}_{nc} - \hat{x}^k_n\|^2 \right)$$

subject to

$$A_n\tilde{x}_{nc} + B_n\tilde{x}_{a(n)c} = b_n \quad , \quad \forall n \in \mathcal{N}_c, \tag{5.17}$$

$$\tilde{x}_{nc} \in \mathcal{X}_n \quad , \quad \forall n \in \mathcal{N}_c. \tag{5.18}$$

Step 2. a) Compute the new group-averaged solution

$$\hat{x}^{k+1}_n \leftarrow \sum_{c\in\mathcal{C}(n)} \tilde{p}_c\hat{p}_{nc}\tilde{x}^k_{nc} / \sum_{c\in\mathcal{C}(n)} \tilde{p}_c\hat{p}_{nc}, \qquad \forall n \in \mathcal{N}_*$$

where $\mathcal{C}(n)$ is the set of groups visiting node n.

b) Update the Lagrange multipliers

$$\tilde{\lambda}_{nc}^{k+1} \leftarrow \tilde{\lambda}_{nc}^{k} + \rho_k(\tilde{x}_{nc}^{k+1} - \hat{x}_{n}^{k+1}), \qquad \forall c \in \mathcal{C},\ n \in \mathcal{N}_c^*.$$

c) Update the penalty parameter using

$$\rho_{k+1} \leftarrow \mu \rho_k \tag{5.19}$$

where $\mu \geq 1$ is a constant that needs to be tuned. Equation (5.19) is the traditional penalty parameter update formula for general ALR methods (Nocedal et Wright, 2006).

d) Stop if

$$\zeta_k \leftarrow \frac{1}{T} \sum_{c \in \mathcal{C}} \tilde{p}_c \sum_{n \in \mathcal{N}_c^*} \|\tilde{x}_{nc}^{k+1} - \hat{x}_{n}^{k+1}\|^2 < \epsilon. \tag{5.20}$$

Otherwise, return to step 1. The stopping criterion ϵ is a positive constant. The metric ζ_k measures the violation level of RNACs at iteration k. In this problem, all decision variables are continuous and all constraints and the objective function are convex. Therefore, the PHA is an exact solution method for this problem. Rockafellar et Wets (1991) presented a convergence analysis for the PHA.

5.4 Scenario set partitioning

5.4.1 Optimal scenario set partitioning problem (OSPP)

The aim of the optimal scenario set partitioning problem (OSPP) is to build a partition of Ω that minimizes the total number of NACs on which an ALR must be applied. Each scenario group Ω_c must not contain more than $N_{\max}$ scenarios to ensure that the size of all subproblems is manageable. The parameter $N_{\max}$ must be chosen to ensure that it does not exceed the computer's available memory. The OSPP is formulated as the following problem

$$\min \sum_{c \in \mathcal{C}} m\,|\mathcal{N}_c^*| \tag{5.21}$$

subject to

$$|\Omega_c| \leq N_{\max} \quad , \quad \forall c \in \mathcal{C} \tag{5.22}$$

$$\bigcup_{c \in \mathcal{C}} \Omega_c = \Omega \quad , \tag{5.23}$$

$$\Omega_c \cap \Omega_d = \emptyset \quad , \quad \forall c, d \in \mathcal{C}, c \neq d \tag{5.24}$$

The objective function (5.21) to be minimized is the total number of RNACs linking groups $c \in \mathcal{C}$ and m is a positive constant representing the number of decision variables in x_t. The constraints (5.22) ensures that all groups cannot contain more than $N_{\max}$ scenarios. The constraints (5.23) ensure that all scenarios are assigned to a particular group. The constraints (5.24) ensure that all subsets Ω_c are disjoint.

5.4.2 Heuristic partitioning method

The OSPP is a combinatorial optimization problem. In fact, the number of feasible partitions grows very rapidly with the number of scenarios in Ω. Even if the size of Ω is moderate (e.g. 100–1,000 scenarios), the OSPP cannot be solved by exhaustive enumeration of all possible partitions. To our knowledge, no exact or heuristic optimization methods have been proposed in the literature for solving this problem.

In this section, we propose a new heuristic method designed to solve the OSPP. The Algorithm 1 summarizes all steps of the scenario set partitioning heuristic (SSPH). This algorithm receives a general scenario tree $\mathcal{T}$ and the parameter $N_{\max}$ which represent the maximal number of scenarios that can be contained in a single group. The proposed method chooses how many groups are required and returns Ω_c, $\tilde{p}_c$ and $\hat{p}_{nc}$ for each $c \in \mathcal{C}$.

Algorithm 1 builds a finite number of groups c sequentially for $c = 1, 2, ..., |\mathcal{C}|$ by selecting a different reference node $\tilde{n}$ among the set of candidate nodes $\tilde{\mathcal{N}}$. The algorithm starts by building group $c = 1$ and only the root node 0 is contained in $\tilde{\mathcal{N}}$. At each iteration of the main `while` loop, the algorithm selects a new reference node $\tilde{n}$ and removes it from $\tilde{\mathcal{N}}$. The node $\tilde{n}$ chosen is the one that is visited by most scenarios in the tree. If node $\tilde{n}$ is visited by more scenarios than is allowed in a single group $N_{\max}$, then this node is not the reference node of current group c to be constructed and all children nodes of $\tilde{n}$ are added to the set $\tilde{\mathcal{N}}$. Otherwise, the node $\tilde{n}$ is the reference node of group c, Ω_c is defined by all scenarios visiting node $\tilde{n}$, the probability of group c corresponds to the occurrence probability of n and the conditional probability of each node $n \in \mathcal{N}_c$ visited by $\omega \in \Omega_c$ is computed as follow :

- The conditional probability of tree nodes $n \in \mathcal{D}(\tilde{n})$ that are located downstream of $\tilde{n}$ is computed using equation (5.12).
- The conditional probability of tree nodes $n \in \mathcal{K}(\tilde{n})$ that are located upstream of $\tilde{n}$ (including $\tilde{n}$ itself) is equal to one.

For the example shown on Fig. 5.3, $\mathcal{D}(4) = \{7, 8, 12, 13, 14\}$ and $\mathcal{K}(4) = \{0, 2, 4\}$. The algorithms continues as long as the set $\tilde{\mathcal{N}}$ is non-empty.

We illustrate how Algorithm 1 works by applying it on the scenario tree shown on Fig.

Algorithm 1 Scenarios set partitioning heuristic (SSPH).

$c \leftarrow 1, \tilde{\mathcal{N}} \leftarrow \{0\}$
while $\tilde{\mathcal{N}} \neq \emptyset$ **do**
  $\tilde{n} \leftarrow \arg\max\{|\Omega(n)| : n \in \tilde{\mathcal{N}}\}$
  $\tilde{\mathcal{N}} \leftarrow \tilde{\mathcal{N}} - \{\tilde{n}\}$
  if $|\Omega(\tilde{n})| > N_{\max}$ **then**
    $\tilde{\mathcal{N}} \leftarrow \tilde{\mathcal{N}} \cup \Delta(\tilde{n})$
  else
    $\Omega_c \leftarrow \Omega(\tilde{n})$
    $\tilde{p}_c \leftarrow p_{\tilde{n}}$
    for $n \in \mathcal{D}(\tilde{n})$ **do**
      $\hat{p}_{nc} \leftarrow p_n/\tilde{p}_c$
    end for
    for $n \in \mathcal{K}(\tilde{n})$ **do**
      $\hat{p}_{nc} \leftarrow 1$
    end for
    $c \leftarrow c + 1$
  end if
end while

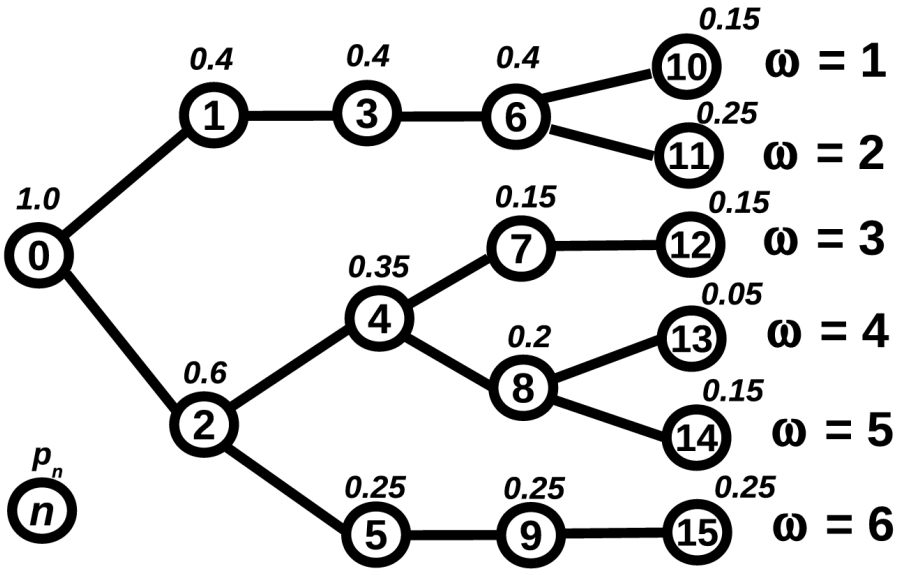

Figure 5.3 Example of a scenario tree.

5.3 with $N_{\max} = 3$. The algorithm returns three groups as shown on Fig. 5.4. The probability

of each group is $\tilde{p}_1 = 0.35$, $\tilde{p}_2 = 0.4$ and $\tilde{p}_3 = 0.25$. The following NACs must be formulated explicitly

$$\tilde{x}_{0,1} = \tilde{x}_{0,2} = \tilde{x}_{0,3} = \hat{x}_0$$

$$\tilde{x}_{2,1} = \tilde{x}_{2,3} = \hat{x}_2.$$

The following intermediary results are obtained in five iterations of the `while` loop

1. $\tilde{\mathcal{N}} = \{0\}$ and, consequently, $\tilde{n} = 0$ is selected and removed from $\tilde{\mathcal{N}}$. The selected node is visited by all scenarios ($\Omega(\tilde{n}) = \{1, ..., 6\}$). Because, $|\Omega(\tilde{n})| = 6 > 3$, the children nodes $\Delta(0) = \{1, 2\}$ are added to $\tilde{\mathcal{N}}$.
2. $\tilde{\mathcal{N}} = \{1, 2\}$. Node $\tilde{n} = 2$ is selected because $|\Omega(2)| = 4 > 2 = |\Omega(1)|$. Because $|\Omega(2)| = 4 > 3$, the children nodes $\Delta(2) = \{4, 5\}$ are added to $\tilde{\mathcal{N}}$.
3. $\tilde{\mathcal{N}} = \{1, 4, 5\}$. Node $\tilde{n} = 4$ is selected and removed from $\tilde{\mathcal{N}}$ because $|\Omega(4)| = 3 > 2 = |\Omega(1)| < 1 = |\Omega(5)|$. Because $|\Omega(4)| = 3 \leq 3$, group $c = 1$ is defined by scenarios $\Omega_1 = \{3, 4, 5\}$ and has a total probability $\tilde{p}_1 = 0.35$. The sets $\mathcal{K}_1 = \{0, 2, 4\}$ and $\mathcal{D}_1 = \{7, 8, 12, 13, 14\}$ are formed. The probability of each node in $c = 1$ is $\hat{p}_{0,1} = \hat{p}_{2,1} = \hat{p}_{4,1} = 1$, $\hat{p}_{7,1} = \hat{p}_{12,1} = \hat{p}_{14,1} = 0.15/0.35$, $\hat{p}_{8,1} = 0.2/0.35$ and $\hat{p}_{13,1} = 0.05/0.35$. We now start constructing group $c = 2$.
4. $\tilde{\mathcal{N}} = \{1, 5\}$. Node $\tilde{n} = 1$ is selected and removed from $\tilde{\mathcal{N}}$ because $|\Omega(1)| = 2 > 1 = |\Omega(5)|$. Because $|\Omega(1)| = 2 \leq 3$, group $c = 2$ is defined by the scenarios $\Omega_1 = \{1, 2\}$ and has a total probability $\tilde{p}_2 = 0.4$. The sets $\mathcal{K}_2 = \{0, 1, 3, 6\}$ and $\mathcal{D}_2 = \{10, 11\}$. The probability of each node in $c = 2$ is $\hat{p}_{0,2} = \hat{p}_{1,2} = \hat{p}_{3,2} = \hat{p}_{6,2} = 1$, $\hat{p}_{10,2} = 0.15/0.4$ and $\hat{p}_{10,2} = 0.25/0.4$. We now start constructing the group $c = 3$.
5. $\tilde{\mathcal{N}} = \{5\}$. Node $\tilde{n} = 5$ is selected and removed from $\tilde{\mathcal{N}}$. Because $|\Omega(5)| = 1 \leq 3$, group $c = 3$ is defined by the scenarios $\Omega_1 = \{5, 6\}$ and has a total probability $\tilde{p}_3 = 0.25$. The sets $\mathcal{K}_3 = \{0, 2, 5\}$ and $\mathcal{D}_3 = \{15\}$ are formed. The probability of each node in $c = 3$ is $\hat{p}_{0,3} = \hat{p}_{2,3} = \hat{p}_{5,3} = \hat{p}_{9,3} = \hat{p}_{15,3} = 1$. $\tilde{\mathcal{N}} = \emptyset$. This is the last iteration because $\tilde{\mathcal{N}} = \emptyset$.

5.5 Numerical experiment

We apply the PHA described in subsection 5.3.4 on an hydroelectric reservoir management problem with stochastic inflows. This problem is formulated as a particular case of the general mathematical program $\mathcal{P}$. Hydrologic uncertainty is modeled by a finite scenario tree. Two different scenario set partitioning methods are compared. In the first method, we apply the SSPH described by Algorithm 1 using different values of $N_{\max}$. The second method builds a random partition of Ω where all groups contains N scenarios.

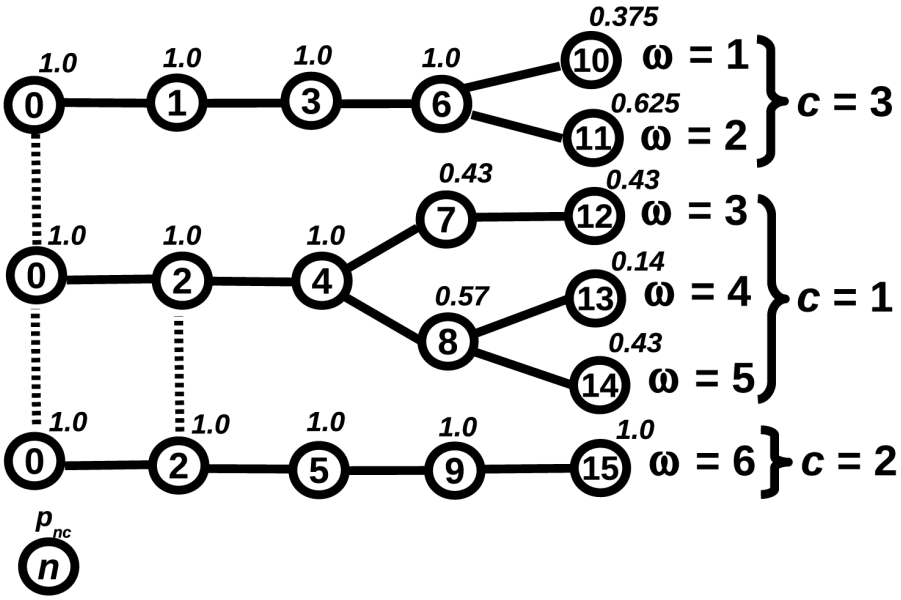

Figure 5.4 Subtrees.

5.5.1 Optimization problem statement

We consider an hydroelectricity producer that operates I hydro plants and J interconnected reservoirs over a T-period planning horizon. The objective function to be maximized is the expected value of

$$\sum_{t=1}^{T}\sum_{i=1}^{I} P_{it}\Delta t + \sum_{j=1}^{J} \alpha_j (v_{jT} - \underline{v}_j) \qquad \text{(MWh)} \tag{5.25}$$

where P_{it} (MW) is the power output of hydro plant i during time period t, Δt (hours) is the time step, v_{jT} (hm^3) is the volume of water stored in reservoir j at the end of time period T, $\underline{v}_j$ (hm^3) is the minimum storage of reservoir j and α_j (MWh/hm^3) is the production factor of reservoir j. The objective function (5.25) contains two parts. The first part represents the amount energy generated by all hydro plants $i = 1, ..., I$ during time periods $t = 1, ..., T$. The second part represents the amount of potential energy stored in reservoirs $j = 1, ..., J$ at the end time period $t = T$.

We assume that the power output P_{it} of hydro plant i during time period t is a concave and piecewise linear function of the turbined outflow q_{it} (m^3 s^{-1}) at i during t and of the volume of water $v_{j(i)t}$ (hm^3) stored in the reservoir $j(i)$ located immediately upstream of i at

the end of t. This assumption enables us model head and generation efficiency variations at hydro plants. Fig. 5.5 shows an illustrative example of a unidimensional production function defined by two pieces $h \in \{1, 2\}$. In the optimization model, the relationship between P_{it}, q_{it} and $v_{j(i)t}$ is represented by the following linear inequality constraints

$$P_{it} \leq \gamma_{ih}^{0} + \gamma_{ih}^{1} q_{it} + \gamma_{ih}^{2} v_{it}, \qquad \forall i, t, h. \tag{5.26}$$

where $\gamma_{ih}^{0}, \gamma_{ih}^{1}, \gamma_{ih}^{2}$ are the linear coefficients of piece h.

The volume of water stored in reservoirs $j = 1, ..., J$ at time periods $t = 1, ..., T$ evolves from a known initial state v_{j0} according to

$$v_{jt} = v_{j,t-1} + \left(\sum_{u \in \mathcal{U}(j)} Q_{ut} - Q_{jt} + \mathcal{I}_{jt} \right) \beta \Delta t, \qquad \forall j, t \tag{5.27}$$

where $\mathcal{U}(j)$ is a set that contains all reservoirs that are located immediately upstream of reservoir j, Q_{jt} ($\mathrm{m}^3\ \mathrm{s}^{-1}$) is the controlled outflow of reservoir j during t, $\beta = 0.0036$ is a constant used for converting flow units into volumetric units and $\mathcal{I}_{jt}$ ($\mathrm{m}^3\ \mathrm{s}^{-1}$) is a random parameter representing the intensity of natural inflows in reservoir j during t. The controlled outflow of reservoir j during t is defined as follows

$$Q_{jt} := s_{jt} + \sum_{i \in I(j)} q_{it} \qquad (\mathrm{hm})^3$$

where s_{jt} ($\mathrm{m}^3\ \mathrm{s}^{-1}$) is the spilled outflow of j during t and $I(j)$ is a set that contains all hydro plants connected directly to reservoir j.

All decision variables should also satisfy the following box constraints

$$\underline{v}_j \leq v_{jt} \leq \overline{v}_j, \qquad \forall j, t \tag{5.28}$$

$$\underline{s}_j \leq s_{jt} \leq \overline{s}_j, \qquad \forall j, t \tag{5.29}$$

$$\underline{P}_i \leq P_{it} \leq \overline{P}_i, \qquad \forall i, t \tag{5.30}$$

$$\underline{q}_i \leq q_{it} \leq \overline{q}_i, \qquad \forall i, t. \tag{5.31}$$

where $(\underline{v}_j, \underline{s}_j, \underline{P}_i, \underline{q}_i)$ and $(\overline{v}_j, \overline{s}_j, \overline{P}_i, \overline{q}_i)$ are parameters representing the lower and upper bounds on decision variables $(v_{jt}, s_{jt}, P_{it}, q_{it})$, respectively.

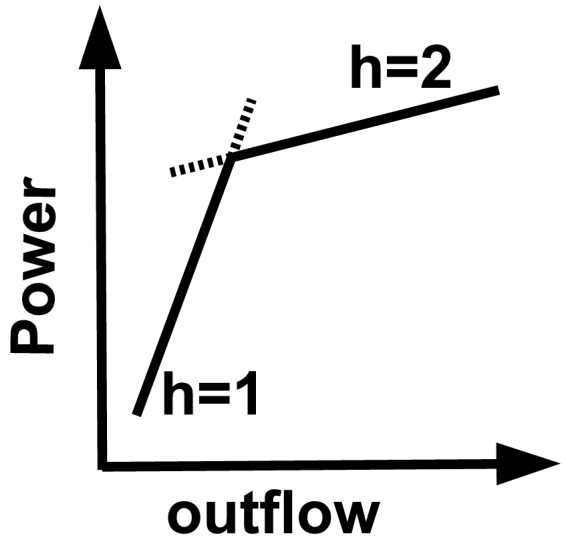

Figure 5.5 Hydroelectricity generation function.

This stochastic optimization problem is a particular case of the general mathematical formulation $\mathcal{P}$ with random right-hand side vectors $b_t(\xi_t)$ at $t = 1, ..., T$. Each component of random vectors

$$\xi_t := (\mathcal{I}_{jt})$$

represents the intensity of natural inflows in a particular reservoir j at the corresponding time period t. The matrices A_t, B_t and cost functions g_t are deterministic and each decision vector is defined as follows

$$x_t := (P_{it}, q_{it}, v_{jt}, s_{jt}).$$

The cost functions at $t = 1, ..., T-1$ are defined as follows

$$g_t(x_t) := -\sum_{i=1}^{I} P_{it}\Delta t.$$

The cost function at $t = T$ is

$$g_T(x_T) := -\sum_{i=1}^{I} P_{iT}\Delta t - \sum_{j=1}^{J} \alpha_j(v_{jT} - \underline{v}_j).$$

The water balance equations (5.27) corresponds to dynamic constraints (5.5). The linear inequality (5.26) and box constraints (5.28)–(5.31) correspond to static constraints (5.6).

5.5.2 Experimental set-up

We test our method on a power system containing $I = 4$ hydro plants and $J = 4$ reservoirs. The power system considered in this experiment has an installed capacity of 1,572 MW and is inpired by a similar reservoir system located in Québec, Canada. The reservoir system has a total storage capacity of 3,710 hm^3. Fig. 5.6 shows the structure of the hydrosystem. The characteristics of each hydro plant and reservoir are summarized on Tables 5.1 and 5.2, respectively. Three hyperplanes ($h = 1, 2, 3$) are used to describe each concave and piecewise linear hydroelectric generation functions. The planning horizon is discretized in $T = 52$ time periods and a time step of $\Delta t = 168$ hours is used. The stopping condition used in this experiment is $\epsilon = 10^{-3}$.

We implemented the PHA in object-oriented C++ using the 12.5.1 version of the ILOG CPLEX and Concert libraries. Subproblems were solved using the barrier solver in parallel mode. All the numerical results presented in this paper were obtained using a personal computer running on Ubuntu 12.04 with a AMD Phenom II X6 2.8 GHz processor and 6 GB of RAM.

Tableau 5.1 Characteristics hydro plants

Plant	Maximum power output (MW)	Maximum turbined outflow ($\text{m}^3\ \text{s}^{-1}$)
1	259	315
2	404	375
3	647	467
4	262	485

Tableau 5.2 Characteristics reservoirs

Reservoir	Minimum storage (hm^3)	Maximum storage (hm^3)	Initial storage (hm^3)	Production factor (MWh/hm^3)
1	952	2,710	1,831	1,091
2	1,403	1,878	1,840	872
3	2,260	3,720	3,645	562
4	129	147	144	158

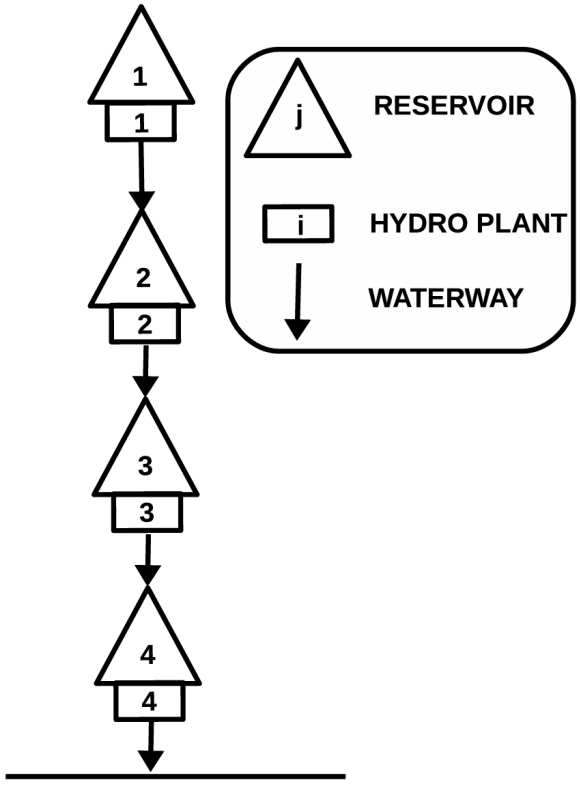

Figure 5.6 Reservoir system.

5.5.3 Scenario tree

In this experiment, we represent the hydrological stochastic process $\{\xi_t : t = 1, ..., T\}$ using a scenario tree containing 500 scenarios and 16,275 nodes. The tree was constructed with the backward algorithm of the SCENRED2 package using 1,000 and 10,000 synthetically generated time series. This software is part of the General Algebraic Modeling System (GAMS) version 23.9.3. The time series were generated using a MPAR(1) stochastic model which was tuned using historical data covering the period 1962–2003.

5.6 Results

5.6.1 Partitioning schemes

Table 5.3 shows the different partitioning schemes that were obtained using the SSPH with different values of $N_{\max}$. The scheme S1 was obtained using $N_{\max} = 1$ scenario per group. Therefore it corresponds to the classical scenario-decomposition scheme. We observe

that the total number of RNACs decreases substantially as $N_{\max}$ increases. Increasing $N_{\max}$ from 10 to 100 decreases the number of RNACs by 95.4 %.

Tableau 5.3 Optimized partitioning schemes.

Scheme	$N_{\max}$	Number of groups	Number of RNACs
S1	1	500	52,696
S2	10	84	2,664
S3	100	12	120

Table 5.4 shows different random partitioning schemes which were obtained using different values of N. The schemes R1a, R1b and R1c are three different replications obtained by partitioning the scenario tree into 50 groups of $N = 10$ scenarios. The average number of RNACs is 17.3 times larger in random schemes than in S2. The schemes R2a, R2b and R2c are three different replications obtained by partitioning the scenario tree into 5 groups of $N = 100$ scenarios. The average number of RNACs is 256 times larger in random schemes than in S3. We observe that increasing the size of scenario groups from $N = 10$ to $N = 100$ decreases average number of RNACs by 33.2 %.

Tableau 5.4 Random partitioning schemes.

Scheme	N	Number of groups	Number of RNACs
R1a	10	50	45,688
R1b	10	50	46,100
R1c	10	50	46,184
R2a	100	5	30,880
R2b	100	5	30,920
R2c	100	5	30,388

5.6.2 Evaluation of partitioning schemes

Table 5.5 shows the results obtained using the PHA with the optimized partitioning schemes that are shown on Table 5.3. These results were obtained using $\rho_0 = 10^{-4}$ and $\mu = 1.2$. We observe that the total running time of the PHA decreases substantially as $N_{\max}$ increases. This improvement is explained by a reduction in the running time per iteration and by a reduction in the number of iterations. The total running time is 7.5 times faster when $N_{\max}$ is increased from 1 to 10 scenarios. The total running time is 54 times faster when

$N_{\max}$ is increased from 10 to 100 scenarios.

Tableau 5.5 Results obtained using optimized partitioning schemes.

Scheme	Iterations	Objective (TWh)	Total time (minutes)	Time per iteration (seconds)
S1	33	10.4782577	122.6	222.9
S2	21	10.4823670	16.3	46.6
S3	1	10.4823916	0.3	17.2

Table 5.6 shows the results obtained using the PHA with the random partitioning schemes shown on Table 5.4 with $\rho_0 = 10^{-4}$ and $\mu = 1.2$. On average, increasing N from 10 to 100 doubled the total running time of the algorithm. This is mainly explained by an increase in the running time per iteration. Also, the number of iterations increased slightly when N increased from 10 to 100. Overall, using a random partitioning method leads to much higher running time than when the SSPH was used. When $N = N_{\max} = 10$, the running time is 44 % higher when a random partition is used in comparison with the SSPH. When $N = N_{\max} = 100$, the running time is 161 times higher when a random partition is used in comparison with the SSPH.

Tableau 5.6 Results obtained with random partitioning schemes.

Scheme	Iterations	Objective (TWh)	Total time (minutes)	Time per iteration (seconds)
R1a	34	10.4814887	23.3	41.1
R1b	34	10.4814797	23.3	41.1
R1c	35	10.4817173	23.9	41.0
R2a	36	10.4821520	46.6	77.7
R2b	38	10.4821276	49.0	77.4
R2c	38	10.4821188	49.2	77.6

5.6.3 Sensitivity to ρ_0 and μ

Table 5.7 shows the numerical results obtained using different values for parameters $\rho_0 \in \{10^{-4}, 10^{-3}\}$ and $\mu \in \{1.1, 1.2, 1.3\}$ using the partitioning schemes S1, S2 and S3. These parameters are used in equation 5.19 to update the penalty parameter ρ_k at each iteration k. We observe that the number of iterations required to satisfy all RNACs and the running time

decrease as ρ_0 and μ increase. This is mainly due to the fact that the penalty parameter weighs the violations of RNACs in subproblems objective function. We also observe that the PHA becomes less sensitive to the choice of μ and ρ_0 when $N_{\max}$ increases. The traditional scenario-decomposition scheme S1 ($N_{\max} = 1$) required between 17 and 51 iterations depending on which values were used for μ and ρ_0. In comparison, the scheme S2 required 10–33 iterations for the same range of parameter values. The scheme S3 required only one iteration, no matter which ρ_0 and μ were used.

Tableau 5.7 Sensitivity to ρ_0 and μ.

Scheme	ρ_0	μ	Iterations	Objective (TWh)	Total time (minutes)	Time per iteration (seconds)
S1	10^{-4}	1.10	51	10.4786199	187.9	221.0
S1	10^{-4}	1.20	33	10.4782577	122.6	222.9
S1	10^{-4}	1.30	26	10.4779222	97.0	223.8
S1	10^{-3}	1.10	28	10.4780311	104.1	223.1
S1	10^{-3}	1.20	20	10.4772046	74.3	222.9
S1	10^{-3}	1.30	17	10.4765857	62.9	222.1
S2	10^{-4}	1.10	33	10.4823691	24.9	45.2
S2	10^{-4}	1.20	21	10.4823670	16.3	46.6
S2	10^{-4}	1.30	17	10.4823587	12.8	45.0
S2	10^{-3}	1.10	13	10.4823524	9.8	45.1
S2	10^{-3}	1.20	11	10.4823410	8.2	44.7
S2	10^{-3}	1.30	10	10.4823319	7.4	44.6
S3	10^{-4}	1.10	1	10.4823916	0.3	17.3
S3	10^{-4}	1.20	1	10.4823916	0.3	17.2
S3	10^{-4}	1.30	1	10.4823916	0.3	17.2
S3	10^{-3}	1.10	1	10.4823916	0.3	17.6
S3	10^{-3}	1.20	1	10.4823916	0.3	17.5
S3	10^{-3}	1.30	1	10.4823916	0.3	17.1

5.7 Conclusions

In this article, we considered an enhanced version of the PHA for solving MSPs that are based on a ST representation of random parameters. The enhanced PHA is based on a multi-scenario decomposition scheme where each subproblem is a MSP defined on a scenario group. We proposed a new heuristic method to partition the scenario set into disjoint subsets (groups) to minimize the number of RNACs. We demonstrated using a numerical experiment that using this partitioning method reduces substantially the running time of the PHA. The

proposed approach was applied on an hydroelectric reservoir management problem with uncertain natural inflows. We compared the performance of the PHA using different partitioning schemes which were built using our approach with a random partitioning method. We also tested the sensitivity of the PHA to the penalty parameter choice for different decomposition schemes. Our numerical results demonstrate that minimizing the number of RNACs leads to much lower running times compared with a random partitioning scheme. Increasing the size of subproblems reduces the number of iterations and reduces the running time per iteration. The proposed approach also decreases the sensitivity of the PHA to the penalty parameter choice.

CHAPITRE 6

ARTICLE 3 : THE EXTENDED L-SHAPED METHOD FOR MID-TERM PLANNING OF HYDROELECTRICITY GENERATION

Pierre-Luc Carpentier[1,2], Michel Gendreau[1,2] and Fabian Bastin[1,3]

[1]Centre interuniversitaire de recherche sur les réseaux d'entreprise, la logistique et le transport (CIRRELT), Montréal, Québec, Canada.

[2]Department of Mathematics and Industrial Engineering, École Polytechnique de Montréal, Montréal, Québec, Canada.

[3]Department of Computer Science and Operations Research, University of Montreal, Montréal, Québec, Canada.

Soumis le 26 juin 2013 au IEEE Transactions on Power Systems

Abstract

In this paper, we propose a new approach for solving the mid-term production planning problem (MTPP) when power systems containing high-capacity (seasonal, multiannual) hydroelectric reservoirs (HCHRs) are considered. For such systems, the mid-term planning horizon typically covers several tens of time periods (weeks) and many external parameters (e.g. inflows, demand, price) can be uncertain. Most random parameters are characterized by a continuous probability distribution which is challenging to represent accurately in stochastic optimization models. A conventional approach to model continuously-distributed random parameters in numerical models consists in replacing the original continuous distribution by a discrete distribution possessing a finite number of realizations (scenarios). This type of approximation leads to a scenario tree (ST) representation of uncertainty. Conventional ST-based decomposition methods (STBMs) (e.g. progressive hedging, nested Benders) are typically used to solve the deterministic equivalent program (DEP). Unfortunately, the size of the DEP grows exponentially with the branching level (BL) used in the ST and, when HCHRs are considered, conventional STBMs can only be used if the BL is quite low (1–2

branch per node). Unfortunately, this corresponds to a very coarse approximation of reality given the high complexity of the underlying distribution. Our approach is designed specifically to consider a higher BL of STBMs when managing high-capacity systems over an extended horizon. To achieve this, we assume that the stochastic process driving random parameters has a memory loss at time period τ where $\tau \gg 1$. The memory loss approximation (MLA) that we propose is well-suited for applications where the MTPP is solved to find a first-period (week) primal (reservoir release) or dual solution (marginal water value) to be used into highly detailed short-term scheduling models. Our approach decreases substantially the amount of memory required by the ST. The proposed approach is tested on an hydroelectric power system in Québec, Canada for a 104-week planning horizon.

6.1 Introduction

Because of their storage capacity and high complexity, large hydroelectric power systems are usually managed over a wide range of time scales. The planning and scheduling of hydroelectricity generation is typically divided in short-, mid- and long-term optimization problems Wallace et Fleten (2003). In this paper, we consider the mid-term planning problem (MTPP) which usually covers several months to a few years, depending on the operational context. Mid-term optimization models (MOMs) are often used by system managers to estimate the current marginal water values (MWVs) or to find weekly water release targets for large reservoirs. These outputs are typically used as an input parameters into highly detailed (e.g. with hourly time steps or less, fixed costs and time delays for start-up/shut-down, water delays, forbidden operating zones, spinning reserves...) short-term (1–10 days) scheduling models (Shawwash *et al.*, 2000). In this type of application, system managers only need an optimal solution at the first time period ($t = 1$) given the current system state (water level in reservoirs, observations/forecast of inflows, price or power demand). The solution at $t = 1$ is generally updated frequently using newly-available information. Another practical use of MOMs is to compute descriptive statistics (mean, variance, quantiles) characterizing future operating conditions (e.g. costs, water level, spillage, available power...) for different possible system configurations. This type of application typically occurs in the context of investment or maintenance planning studies. In such case, system managers usually need a full closed-loop operating policy (feedback control law) returning an optimal control for any possible system state at all time periods $t = 1, ..., T$. The resulting closed-loop policy can be simulated using different historical or synthetically-generated time series characterizing all random parameters (e.g. inflows, power demand, price).

The main source of complexity of MTPPs is due to the high uncertainty of the decision environment. In most applications, several input parameters are random most of which are characterized by a continuous (joint) probability distribution which is quite challenging to represent in MOMs. In general, several modeling assumptions have to be made to keep MOMs tractable numerically :

- The mid-term planning horizon is discretized coarsely using weekly or monthly time steps.
- Generating units aggregated as power plants.
- The power output of hydro plants is modeled by concave and piecewise linear functions (Diniz et Maceira, 2008).
- Forbidden zones due to mechanical vibration or cavitation are neglected.
- All fixed costs and time delays associated with the start-up or shut-down of generating units are neglected.
- Short-term (hourly and daily) variations of power demand and energy prices are usually neglected or, in some models, these variations are represented in a simplified manner using a discretized load- or price-duration curve with 3–5 possible values.

Deterministic optimization models are sometimes used for solving MTPPs. These models are built on the simplifying assumption that all input parameters are known precisely over the entire planning horizon. This assumption reduces drastically the computational burden, but often lead to poor (high cost, spillage) solutions as pointed out by Philbrick et Kitanidis (1999). Stochastic optimization models are commonly used for solving MTPPs. Many stochastic optimization methods were proposed over the past decades for managing hydroelectric reservoir systems over the mid-term planning horizon. A comprehensive literature review of these methods can be found in Yeh (1985), Labadie (2004) and Rani et Moreira (2010). Finding the best possible approach for solving a particular MTPP can be quite difficult given the large number of existing methods. In general, each method has its own advantages and limitations. No method was proven to be superior for all possible types of real-world situations. In general, the applicability of a particular method depends on type of solution needed (e.g. primal or dual solution at the first time period, optimal policy at all time periods), on the size (e.g. storage capacity, number of controllable components) and complexity level of the controlled system, on the operational constraints to be satisfied (e.g. minimum load requirement, transmission network, maintenance, energy trading...) and on the characteristics of uncertainty (e.g. number of random parameters, spatiotemporal correlation effects, variability...).

Solution methods based on the dynamic programming (DP) principle (Bellman, 1957) are widely used for managing hydroelectric reservoir systems Yakowitz (1982); Nandalal et Bogardi (2007). These methods return a close-loop policy and work by constructing a recourse (or Bellman, expected cost-to-go) function $\mathcal{Q}_t(x_t)$ at $t = 1, ..., T$ of the planning horizon. Each of these functions returns the optimal expected cost at $t, t+1, ..., T$ conditional to the current system state vector x_t. State vectors x_t must contain all the necessary informations (e.g. reservoir level, observations or forecast for random parameters) to compute future expected costs. Highly complex hydroelectric reservoir systems (nonlinear, non-convex) can be managed using the discrete stochastic DP (DSDP) methods Tejada-Guibert *et al.* (1995). Unfortunately, the computational complexity of the DSDP algorithms grows exponentially with the number of dimensions of state and control vectors because of the so-called *curse of dimensionality*. For this reason, DSDP methods are only applicable on low dimensional systems (with less than 5 state variables). Dimensionality problems associated with DSDP methods can be mitigated using Approximate DP (ADP) algorithms Powell (2011). The stochastic dual DP (SDDP) algorithm Pereira et Pinto (1991) is a particular type of ADP method which was used repeatedly for solving linear MTPPs. This method was first developed in Brazil Maceira *et al.* (2008). Subsequently, the SDDP was also applied in Norway Rotting et Gjelsvik (1992); Gjelsvik *et al.* (2010) and in other regions of the world (Tilmant et Kelman, 2007; Tilmant *et al.*, 2008). With this method, the *curse of dimensionality* associated with state and control spaces is avoided since the method is based on continuous spaces. This method works by constructing a convex and piecewise linear approximation of $\mathcal{Q}_t$ by generating Benders (1962) cuts.

Another possible type of approach would be to describe random parameters using a finite scenario tree (ST). With this approach, each possible realization (scenario) of the underlying stochastic process corresponds to a path from the tree root to a particular leaf. Fig. 6.1 shows an example of a ST. Different approaches were proposed over the years to construct scenario trees for stochastic programs (Dupačová *et al.*, 2000; Gröwe-Kuska *et al.*, 2003; Heitsch et Römisch, 2005; Latorre *et al.*, 2007). Heitsch et Römisch (2009) proposed a powerful method which enables to construct a ST of prescribed size from a set of historical or synthetically generated time series. Their method is based on the minimization of a probability distance and is based on theoretical results obtained by Heitsch *et al.* (2006). Stochastic programs defined on a ST can be formulated explicitly as a large-scale deterministic equivalent mathematical program (DEP) whose size (number of variables and constraints) is proportional to the dimension of the controlled system. Because of this, ST-based methods (STBMs) are

generally well-suited for managing high dimensional systems containing tens of controllable components (reservoirs, plants). Contrary to DP-based methods, STBMs do not return a complete operating policy for the power system. Indeed, STBMs return a finite decision tree whose branching structure is identical to the ST used to describe random parameters. The decision tree contains an optimal decision at each node. Each node corresponds to a possible system state at a given time period. In general, STBMs are particularly well-suited for practical applications where only the solution at $t = 1$ is needed. The root node solution can be used to fill this need. The main limitation of STBMs is due to the exponential growth of the DEP with the branching level (number of branches per node) of the ST. In most cases, the DEP is too large to be solved directly and decomposition methods are normally used. Different types of decomposition strategies were proposed over the years for solving large-scale DEPs (Sagastizabal, 2012; Birge et Louveaux, 2011). The classical nested Benders decomposition (NBD) (Birge, 1985) uses a node-wise decomposition scheme where each subproblem corresponds to one two-stage problem at a given node. This method was used repeatedly in previous studies (Jacobs *et al.*, 1995; Archibald *et al.*, 1999; M. L. L. dos Santos et Goncalves, 2009). A different version of the NBD was proposed by dos Santos et Diniz (2009). Instead of using the conventional node-wise decomposition scheme, these authors applied a subtree decomposition scheme where each subproblem is defined on several nodes. Küchler et Vigerske (2007) proposed an extension of the NBD for considering recombining STs. The progressive hedging (PH) algorithm (Rockafellar et Wets, 1991) is another STBM which was used in past studies for managing hydroelectric power systems (Goncalves *et al.*, 2011; Carpentier *et al.*, 2013b). This method works by applying an augmented Lagrangian relaxation on non-anticipativity constraints of the DEP.

Representing the stochastic process $\{\xi_t : t = 1, ..., T\}$ using a ST enables representing multi-lag time-autocorrelation effects without increasing the computational burden associated of the DEP. In general STs, the distribution $\mathbb{P}_t(\cdot|\xi_{t-1}, ..., \xi_1)$ characterizing random vector ξ_t can depend on the entire history $\xi_{t-1}, ..., \xi_1$ of the underlying stochastic process. However, applying conventional STBMs (e.g. NBD or PH algorithms) becomes particularly challenging when the number of time periods T is large. This type of situation typically occurs when the power system to be managed contains high-capacity reservoirs (HCRs) characterized by a seasonal or multiannual drawdown-refill cycle. For such systems, the mid-term planning horizon usually covers tens of time periods, and STBMs can only be applied if a very low branching level (1–2 branch per node) is used. For example, if the horizon covers $T = 52$ periods and if we only consider only three possible outcomes per time period, then the ST would contain 2×10^{24} scenarios and 3×10^{24} nodes.

In this paper, we propose a new modeling assumption designed to consider a higher branching level when T is large. Among the numerous modeling approximations being made in the MTPP, the low branching level of the ST is usually the crudest one given the high complexity (continuous, multidimensional, asymmetric, wide range of possible outcomes, ...) of $\mathbb{P}_t$. To approximate $\mathbb{P}_t$ using a higher discretization level, we assume that the stochastic process $\{\xi_t : t = 1, ..., T\}$ driving random parameters $\xi_t \in \mathbb{R}^m$ has a memory loss at the end of time period $\tau \gg 1$. The memory loss approximation (MLA) is particularly well-suited for real-world applications where only the solution at $t = 1$ is implemented. In reality, the underlying stochastic process that we try to approximate using a ST usually has some time-correlation effects over the entire planning horizon. Despite this, the MLA can still be useful in real-world applications since it enables to better represents the variability of random parameters over the entire planning horizon than if a general ST was used. In this study, we demonstrate how the MLA can be exploited using a two-stage Benders decomposition scheme. The proposed algorithm is an extension of the classical L-Shaped method of Van Slyke et Wets (1969) which was originally designed to solve general two-stage stochastic linear programs with random right-hand side vectors.

In Section II of this paper, we describe the mathematical problem to be solved. Section III describes the extended L-Shaped (ELS) algorithm in details. A numerical experiment performed with this algorithm is presented in Section IV. In Section V, we present the numerical results obtained for this experiment. Comments and conclusions are drawn in Section VI.

6.2 Mid-term planning problem

6.2.1 Problem description

We consider a multicomponent power system controlled over a T-period planning horizon in an uncertain decision environment. The controlled system contains hydro plants connected to hydroelectric reservoirs $j = 1, ..., J$. The system may also contain other controllable components such as thermal generating units and a transmission network with energy trading (selling, purchasing) nodes.

At the beginning of each time period $t = 1, ..., T$, the decision maker must choose a new control vector u_t which minimizes the expected operating cost during remaining time periods $t, t + 1, ..., T$. Each vector u_t contains all the necessary decision variables (e.g. water release, spillage, power flow, trading, thermal generation,...) to operate the entire power system du-

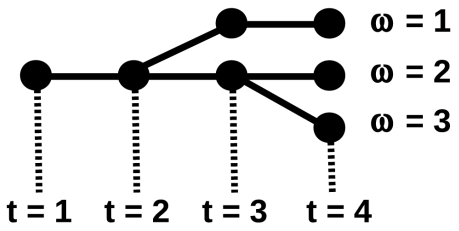

Figure 6.1 Scenario tree.

ring t. The reservoir system state $v_t = (v_{jt})$ (hm^3) evolves linearly from a known initial state $V^{init} = (V_j^{init})$ (hm^3). Each component v_{jt} (hm^3) of state vector $v_t = (v_{jt})$ represents the volume of water stored in reservoir j at the end of t. Each component V_j^{init} (hm^3) of the initial state vector V^{init} represents the volume of water stored in reservoir j at the beginning of $t = 1$.

The cost $g_t(v_t, u_t, \xi_t)$ (\$) during t is a convex and piecewise linear function of u_t, v_t and depends on random vector ξ_t. The value of reservoir storage at the end $t = T$ is returned by a concave and piecewise linear function $\Psi(v_T)$ (\$). Each component of ξ_t represents a random parameter (e.g. load, inflows, prices, failures, ...) during t. The stochastic process $\{\xi_t : t = 1, ..., T\}$ may be time-autocorrelated. Components of ξ_t at a given t may be cross-correlated. Consequently, each random vector ξ_t is characterized by a probability distribution $\mathbb{P}_t(\,\cdot\,|\,\xi_1, ..., \xi_{t-1})$ which depends on the historical values of $\xi_1, ..., \xi_{t-1}$.

6.2.2 Stochastic programming formulation

The MTPP is formulated as the following multiperiod stochastic linear program (MSLP)

$$(\mathcal{P})\ z_* = \min_{u_t, v_t} \mathbb{E}_{\xi_1, ..., \xi_T} \left[\sum_{t=1}^{T} g_t(v_t, u_t, \xi_t) - \Psi(v_T) \right]$$

subject to

$$v_{j0} = V_j^{init} \quad , \quad \forall j \tag{6.1}$$

$$A_t^1(\xi_t)v_t + A_t^2(\xi_t)v_{t-1} + A_t^3(\xi_t)u_t = b_t^1(\xi_t) \quad , \quad \forall t \tag{6.2}$$

$$A_t^4(\xi_t)v_t + A_t^5(\xi_t)v_{t-1} + A_t^6(\xi_t)u_t \geq b_t^2(\xi_t) \quad , \quad \forall t \tag{6.3}$$

$$\underline{u}_t \leq u_t \leq \overline{u}_t,\ \underline{v}_t \leq v_t \leq \overline{v}_t \quad , \quad \forall t \tag{6.4}$$

$$u_t \in \mathcal{F}_t(\xi_1, ..., \xi_t) \quad , \quad \forall t \tag{6.5}$$

where z_* is the optimal expected cost and $\mathbb{E}_{\xi_1,...,\xi_T}[\,\cdot\,]$ is the expectation operator computed with respect to the probability distribution of $\xi_1, ..., \xi_T$. We define Λ_j (\$ hm^{-3}) as the Lagrange multiplier associated with the initial state constraint (6.1) for reservoir j. Λ_j can be interpreted as the MWV for reservoir j at the beginning of the planning horizon. Constraints (6.2)–(6.4) represent static and dynamic physical constraints (e.g. water balance, energy budget, hydroelectricity generation, operating range, ...). Coefficient of matrices $A_t^1(\xi_t), ..., A_t^6(\xi_t)$ and vectors $b_t^1(\xi_t), b_t^2(\xi_t)$ may be random. The non-anticipativity constraints (6.5) ensure that each control vector is only a function of known information $(\xi_1, ..., \xi_t)$. Each set $\mathcal{F}_t$ is defined by non-anticipativity constraints at time t.

6.2.3 Memory loss approximation (MLA)

In this paper, we partition the planning horizon in two consecutive stages. The first stage covers time periods $t = 1, ..., \tau$ and the second stage covers time periods $t = \tau + 1, ..., T$. We assume that the stochastic process $\{\xi_t : t = 1, ..., T\}$ has a memory loss at the end of the first stage ($t = \tau$). In practice, the timing of the memory loss may correspond to a seasonal change (e.g. onset of the spring flood or freezing season). Consequently, the probability distributions $\mathbb{P}_t(\,\cdot\,|\,\xi_{\tau+1}, ..., \xi_{t-1})$ for any $t \geq \tau+1$ may only depend on the outcome of second-stage random vectors $\xi_{\tau+1}, ..., \xi_{t-1}$.

6.2.4 Decomposition scheme

We exploit the MLA to transform $\mathcal{P}$ as follows. Firstly, we split the summation operator in two parts

$$z_* = \min_{x_1,...,x_T} \mathbb{E}_{\xi_1,...,\xi_T} \left[\sum_{t=1}^{\tau} g_t(v_t, u_t, \xi_t) + ... \right.$$
$$\left. ... + \sum_{t=\tau+1}^{T} g_t(v_t, u_t, \xi_t) - \Psi(v_T) \right].$$

Then, the expectation is computed in two steps. The outer expectation is computed with respect to the probability distribution of random vectors $\xi_1, ..., \xi_\tau$. The inner expectation is computed with respect to the probability distribution of random vectors $\xi_{\tau+1}, ..., \xi_T$. The resulting problem is

$$z_* = \min_{x_1,...,x_T} \mathbb{E}_{\xi_1,...,\xi_\tau} \left[\sum_{t=1}^{\tau} g_t(v_t, u_t, \xi_t) + ... \right.$$
$$\left. ... + \mathbb{E}_{\xi_{\tau+1},...,\xi_T} \left[\sum_{t=\tau+1}^{T} g_t(v_t, u_t, \xi_t) \middle| \xi_1, ..., \xi_\tau \right] \right].$$

For general stochastic processes, the inner expectation is conditional to the realization of $\xi_1, ..., \xi_\tau$ because of time-autocorrelation effects. Because of the MLA, this expectation no longer depends on $\xi_1, ..., \xi_\tau$ and the optimization problem becomes

$$z_* = \min_{x_1,...,x_T} \mathbb{E}_{\xi_1,...,\xi_\tau} \left[\sum_{t=1}^{\tau} g_t(v_t, u_t, \xi_t) + ... \right.$$
$$\left. ... + \mathbb{E}_{\xi_{\tau+1},...,\xi_T} \left[\sum_{t=\tau+1}^{T} g_t(v_t, u_t, \xi_t) \right] \right].$$

The minimization operator can be splitted in two parts as follows

$$z_* = \min_{x_1,...,x_\tau} \mathbb{E}_{\xi_1,...,\xi_\tau} \left[\sum_{t=1}^{\tau} g_t(v_t, u_t, \xi_t) + ... \right.$$
$$\left. ... + \min_{x_{\tau+1},...,x_T} \left\{ \mathbb{E}_{\xi_{\tau+1},...,\xi_T} \left[\sum_{t=\tau+1}^{T} g_t(v_t, u_t, \xi_t) \right] : v_\tau \right\} \right]$$

where the outer minimization operator is performed with respect to first-stage decision variables $x_1, ..., x_\tau$. The inner minimization operator is performed with respect to second-stage decision variables $x_{\tau+1}, ..., x_T$ and is conditional to v_τ resulting from the first-stage optimization problem (outer minimization). The outer minimization problem can be transformed into an equivalent master problem

$$z_* = \min \mathbb{E}_{\xi_1,...,\xi_\tau} \left[\sum_{t=1}^{\tau} g_t(v_t, u_t, \xi_t) + \mathcal{Q}(v_\tau) \right]$$

subject to constraints (6.1)–(6.5) for all $t = 1, ..., \tau$. The recourse function is defined as follows

$$\mathcal{Q}(v_\tau) := \min_{x_{\tau+1},...,x_T} \left\{ \mathbb{E}_{\xi_{\tau+1},...,\xi_T} \left[\sum_{t=\tau+1}^{T} g_t(v_t, u_t, \xi_t) \right] : v_\tau \right\}$$

subject to constraints (6.1)–(6.5) for all $t = \tau + 1, ..., T$. We split the expectation operator into an outer and an inner expectation as follows

$$\mathcal{Q}(v_\tau) = \min_{x_{\tau+1},...,x_T} \left\{ \mathbb{E}_{\xi_{\tau+1}} \left[\mathbb{E}_{\xi_{\tau+2},...,\xi_T} \left[\sum_{t=\tau+1}^{T} g_t(v_t, u_t, \xi_t) \,\middle|\, \xi_{\tau+1} \right] \right] : v_\tau \right\}.$$

The outer expectation is computed with respect to $\xi_{\tau+1}$. The inner expectation is computed with respect to $\xi_{\tau+2}, ..., \xi_T$ and is conditional to $\xi_{\tau+1}$. Because $\xi_{\tau+1}$ is revealed to the decision maker at the beginning of $\tau + 1$, we can invert the minimization and expectation operator and the recourse function becomes

$$\mathcal{Q}(v_\tau) = \mathbb{E}_{\xi_{\tau+1}} \left[\min_{x_{\tau+1},...,x_T} \left\{ \mathbb{E}_{\xi_{\tau+2},...,\xi_T} \left[\sum_{t=\tau+1}^{T} g_t(v_t, u_t, \xi_t) \,\middle|\, \xi_{\tau+1} \right] : v_\tau \right\} \right].$$

or, equivalently

$$\mathcal{Q}(v_\tau) = \mathbb{E}_{\xi_{\tau+1}} \left[Q(v_\tau, \xi_{\tau+1}) \right].$$

where

$$(\mathcal{S}) \quad Q(\hat{v}, \xi_{\tau+1}) := \min_{x_{\tau+1},...,x_T} \left\{ \mathbb{E}_{\xi_{\tau+2},...,\xi_T} \left[\sum_{t=\tau+1}^{T} g_t(v_t, u_t, \xi_t) \,\middle|\, \xi_{\tau+1} \right] \right\}.$$

subject to (6.2)–(6.5) for $t = \tau + 1, ..., T$ and to the constraint

$$v_\tau = \hat{v}, \qquad (\lambda)$$

where $\lambda = (\lambda_j)$ (\$ hm^{-3}) is the associated dual vector. The previous stochastic program correspond to the subproblem of our algorithm.

6.2.5 Scenario tree approximation

We assume that each random vector ξ_t has a finite number of possible outcomes at each time period t. All random parameters are assumed to be exogenous to the controlled system. Consequently, the probability distributions $\mathbb{P}_t$ are not influenced by controlled u_t or by v_t.

Making these assumptions enables us to represent the multidimensional stochastic process $\{\xi_t : t = 1, ..., T\}$ using a scenario tree $\mathcal{T}$ which contains a finite set of nodes $\mathcal{N}$. The tree structure is characterized by a function $a(n)$ which returns the ancestor of any node in it. Each node n is associated with a time period $t(n)$. A random vector ξ_n and an occurrence probability $p_n \in (0, 1)$ are defined at each tree node $n \in \mathcal{N}$. Each path from the root $0 \in \mathcal{N}$ to a leaf $\ell(\omega) \in \mathcal{L}$ corresponds to a particular realization (scenario) $\omega \in \Omega$ of the stochastic process. The occurrence probability of a scenario ω is the probability $p_{\ell(\omega)}$ of the associated leaf $\ell(\omega)$.

Because of the MLA, the scenario tree $\mathcal{T}$ possesses a two-stage structure with redundancy at the second stage. The first-stage subprocess $\{\xi_t : t = 1, ..., \tau\}$ is represented by a first-stage subtree $\mathcal{T}_1$ containing nodes $\mathcal{N}_1$. Each leaf in $\mathcal{T}_1$ is an ancestor for the root $r(m)$ of a second-stage subtrees $\mathcal{T}_2^m$ containing nodes $\mathcal{N}_2^m$ where $m \in \mathcal{M} = \{1, ..., M\}$. The master problem and the subproblem are relatively small stochastic programs defined on finite subtrees $\mathcal{T}_1$ and $\mathcal{T}_2^m$ respectively. Therefore, these problems can be formulated as DEPs possessing a finite number of constraints and decision variables. The second-stage expected cost $Q_{\ell m}(\hat{v}) = Q(\hat{v}_\ell, \hat{\xi}_{\tau+1})$ for a particular $\hat{\xi}_{\tau+1} \in \Xi_{\tau+1} = \{\hat{\xi}_{\tau+1}^m : m = 1, ..., M\}$ is a convex and piecewise linear function of $\hat{v}_\ell$ at $\ell \in \mathcal{L}_1$. Therefore, it can be represented using a finite number of cuts

$$Q_{\ell m}(\hat{v}_\ell) \geq \alpha_{hm} + \lambda_{hm}^T \hat{v}_\ell, \qquad \forall h \in \mathcal{H}_m$$

where $\alpha_{hm}, \lambda_{hm}$ represent parameters of cut h, $(\cdot)^T$ is the transpose operator, and $\mathcal{H}_m$ is the set of cuts.

Fig. 6.2 shows a simple example where $T = 6$, $\tau = 3$ and $M = 2$. Fig. 6.3 shows subtree $\mathcal{T}_1$. Fig. 6.4a and 6.4b show subtrees $\mathcal{T}_2^1$ and $\mathcal{T}_2^2$, respectively. The number of tree nodes in $\mathcal{T}_1$, $\mathcal{T}_2^1$ and $\mathcal{T}_2^2$ is much lower than for the original scenario tree $\mathcal{T}$. For example, if subtrees contain N_{nodes} nodes and N_{br} branches, the resulting number of tree nodes needed to represent $\mathcal{T}_1$ and $\mathcal{T}_2^m$ is simply $N_{nodes}(1 + M)$. A total number of nodes necessary to represent the equivalent scenario tree $\mathcal{T}$ would be $N_{nodes}(1 + N_{br}M)$. The reduction factor

$$RF = \frac{1 + M}{1 + N_{br}M}$$

is obtained by dividing the number of nodes in all subtrees by the number of nodes in the full scenario tree. In general, $N_{br}M \gg 1$ and $RF \approx 1/N_{br}$. For example, if each subtree contains 100 branches, the equivalent scenario tree would contain approximately 100 times more nodes.

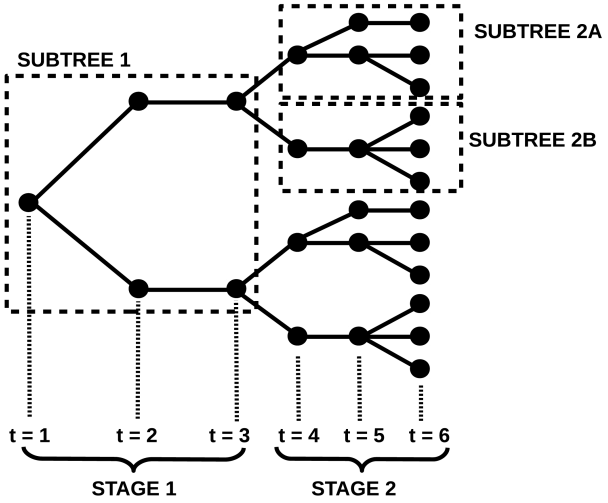

Figure 6.2 Scenario tree $\mathcal{T}$ with memory loss at $\tau = 3$.

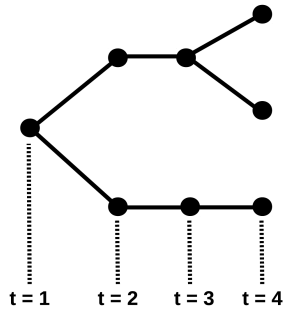

Figure 6.3 Subtree $\mathcal{T}_1$ at the first stage.

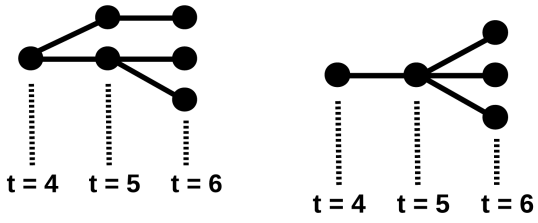

Figure 6.4 Second-stage subtrees $\mathcal{T}_2^m$ with initial condition $m = 1$ (left) and $m = 2$ (right).

6.3 Solution method

6.3.1 Description

Essentially, our method works by constructing iteratively an outer approximation $\tilde{\mathcal{Q}}_{\ell m}$ of the exact recourse functions $\mathcal{Q}_{\ell m}$. These approximations are used in the master problem to find a set of candidate state vectors for each leaf in $\mathcal{T}_1$. Then, a subproblem is solved for each of those newly generated candidate states and for each second-stage subtree $\mathcal{T}_2^m$ in order to generate new Benders cuts. At the next iteration, the newly generated cuts are used to refine the approximation of the recourse function in the master problem.

6.3.2 Extended L-Shaped (ELS) algorithm

The algorithm is initialized with a set $\mathcal{V}_1 \leftarrow \{\hat{v}_\ell^1 : \ell \in \mathcal{L}_1\}$ containing a feasible state vector v_ℓ^1 for each leaf $\ell \in \mathcal{L}_1$ in $\mathcal{T}_1$. Set the lower bound as $\underline{z}_1 \leftarrow \infty$. Set the first-stage cost as $\zeta_1 \leftarrow \infty$. The initial set of cuts $\mathcal{H}_m^k \leftarrow \emptyset$. At each iteration $k = 1, 2, ...$, perform the three following steps :

Step 1. For each $\hat{v}_\ell^k \in \mathcal{V}_k$ and for each $m \in \mathcal{M}$, solve the subproblem associated with $\mathcal{T}_2^m$

$$(\mathcal{S}_{\ell m}^k)\ \theta_{\ell m}^k \leftarrow \min \sum_{n \in \mathcal{N}_2^m} p_n g_n(v_n, u_n) - \sum_{\ell \in \mathcal{L}_2^m} p_\ell \Psi(v_\ell)$$

subject to

$$v_{jr(m)} = \hat{v}_{j\ell}^k, \qquad \forall j \tag{6.6}$$

$$A_n^1 v_n + A_n^2 v_{a(n)} + A_n^3 u_n = b_n^1, \quad \forall n \in \mathcal{N}_2^m \tag{6.7}$$

$$A_n^4 v_n + A_n^5 v_{a(n)} + A_n^6 u_n \geq b_n^2, \quad \forall n \in \mathcal{N}_2^m \tag{6.8}$$

$$\underline{u}_{t(n)} \leq u_n \leq \overline{u}_{t(n)}, \underline{v}_{t(n)} \leq v_n \leq \overline{v}_{t(n)}, \quad \forall n \in \mathcal{N}_2^m \tag{6.9}$$

where $\theta_{\ell m}^k$ is the objective value, and $\lambda_{m\ell}^k$ is the dual vector associated with the initial conditions constraints (6.6). The set $\mathcal{L}_2^m$ contains all leaves in $\mathcal{T}_2^m$. The cost functions are convex and piecewise linear and defined as follows $g_n(v_n, u_n) := g_{t(n)}(u_n, v_n, \xi_n)$. The value function Ψ is concave and piecewise linear. The cost and value functions appear in the objective function and can be represented by a finite number of linear inequality constraints. The function $r(m)$ returns the root node of subtree $\mathcal{T}_2^m$. The objective value and the dual solution of $\mathcal{S}_{\ell m}^k$ are used to generate a new cut

$$Q_{\ell m} \geq \theta_{\ell m}^k + (\lambda_{m\ell}^k)^T (v_\ell - \hat{v}_\ell^k)$$

which must be added to $\mathcal{H}_m^k$ to refine the approximation of the expected cost-to-go function in the master problem.

Step 2. Solve the master problem

$$\underline{z}_k \leftarrow \min \sum_{n\in\mathcal{N}_1} p_n g_n(v_n, u_n) + \sum_{\ell\in\mathcal{L}_1} p_\ell \sum_{m\in\mathcal{M}} p_{r(m)} Q_{\ell m}$$

subject to

$$Q_{\ell m} \geq \alpha_{hm} + \lambda_{hm}^T v_\ell, \quad \forall h, m, \ell \tag{6.10}$$

$$A_n^1 v_n + A_n^2 v_{a(n)} + A_n^3 u_n = b_n^1, \quad \forall n \in \mathcal{N}_1 \tag{6.11}$$

$$A_n^4 v_n + A_n^5 v_{a(n)} + A_n^6 u_n \geq b_n^2, \quad \forall n \in \mathcal{N}_1 \tag{6.12}$$

$$\underline{u}_{t(n)} \leq u_n \leq \overline{u}_{t(n)}, \underline{v}_{t(n)} \leq v_n \leq \overline{v}_{t(n)}, \quad \forall n \in \mathcal{N}_1 \tag{6.13}$$

$$v_{ja(\rho)} = V_j^{init}, \qquad \forall j \tag{6.14}$$

where $Q_{\ell m}$ is a decision variable representing the expected cost at the second-stage for subtree $\mathcal{T}_2^m$ for v_ℓ at leaf $\ell \in \mathcal{L}_1$. The constraints (6.10) are used to represent the convex and piecewise linear expected cost-to-go functions. The solution of the master problem is used to generate a new set of feasible state vectors $\mathcal{V}_{k+1} \leftarrow \{\hat{v}_\ell^{k+1} : \ell \in \mathcal{L}_1\}$ where $\hat{v}_\ell^{k+1}$ are the state vectors at leaves $\ell \in \mathcal{L}_1$. The master problem and subproblems can be solved directly or using some decomposition method such as the NBD or the PH algorithms.

Step 3. Compute an upper bound $\overline{z}_k$ on z_* as follows

$$\overline{z}_k \leftarrow \zeta_{k-1} + \sum_{\ell\in\mathcal{L}_1} p_\ell \sum_{m\in\mathcal{M}} p_{r(m)} \theta_{\ell m}^k$$

where

$$\zeta_{k-1} \leftarrow \sum_{n\in\mathcal{N}_1} p_n g_n(\hat{v}_n^{k-1}, \hat{u}_n^{k-1})$$

is the first stage at the previous iteration and $(\hat{v}_n^{k-1}, \hat{u}_n^{k-1})$ is the corresponding optimal solution at the first stage. The initial value is defined as follows $\zeta_0 = \infty$.

The algorithm stops if

$$(\overline{z}_k - \underline{z}_k)/\left|\underline{z}_k\right| < \epsilon$$

where $\epsilon > 0$ is the stopping criterion and $\underline{z}_k$ is the lower bound on z_*. The lower bound is the objective value of the current master problem. The bound $\underline{z}_k$ is computed whenever the

master problem is solved to optimality. The ELS algorithm is an adaptation of the classical L-Shaped method Van Slyke et Wets (1969) which was originally designed to solve two-stage stochastic programs where each stage corresponds to one time period. Therefore, the ELS algorithm converges to an optimal solution of $\mathcal{P}$.

6.4 Case study

In this experiment, we consider a power producer located in Québec, Canada which operates a power system containing $J = 8$ interconnected reservoirs over $T = 104$ weeks. In this problem, hydroelectricity generation must be coordinated with thermal generation and energy trading. Head and efficiency are taken into account in this problem. The intensity of natural inflows in each reservoir is highly uncertain and characterized by large seasonal and interannual variations. Fig. 6.5 shows the total inflow for the entire system observed over 42 years (1962–2003). The planning horizon begins on January 1^{st} during the freezing season. We assume that the hydrological stochastic process has a memory loss during the second freezing season at week $\tau = 52$. In other words, we assume that the inflow at the second year is independent from the inflow at the first year. It is worth mentioning that our approach could be used in a different context (e.g. after the end of the spring flood). Fig. 6.6 shows a scatter plot comparing the annual volume of natural inflow for each pair of consecutive years in the historical record. This plot shows that the correlation is relatively weak in annual volume.

6.4.1 Controlled system

The controlled power system contains reservoirs $j = 1, ..., 8$, hydroelectric generating stations $i = 1, ..., 10$ and a thermal generating station. Fig. 6.7 illustrates the reservoir network and the location of each generating station. Tables I and II show the physical characteristics of each reservoir and hydro plant. Hydropower generation is assumed to be cost-free. We assume that the marginal cost for thermal generation is constant and equal to 60 \$ MWh^{-1}. We assume that the thermal plant has an infinite production capacity. Fixed costs (start-up/shut-down) are assumed to be negligible.

The producer can sell its energy surpluses on a deregulated market at a known price π_t (\$ MWh^{-1}). The average price is 37.42 \$ MWh^{-1} and the standard deviation is 5.33 \$ MWh^{-1}. We assume that the power producer is a price-taker and that the market has an infinite capacity. No transaction cost are considered in this problem.

Tableau 6.1 Characteristics of reservoirs.

j	Maximal storage (hm^3)	Initial storage (hm^3)	ψ_j (k\$/hm^3)
1	28,224	9,526	34.6
2	326	317	20.2
3	171	170	10.7
4	16	15	3.6
5	2,436	1,325	27.4
6	10,940	6,523	34.4
7	15	14	22.6
8	16	14	8.2

Tableau 6.2 Characteristics of hydroelectric generating stations.

i	Installed capacity (MW)	Head (m)	Number of units
1	1,596	141.8	8
2	1,064	144.5	4
3	1,244	94.19	6
4	1,145	70.11	8
5	184	36.58	3
6	235	37.8	7
7	526	152	2
8	785	120.55	4
9	1,026	143.57	4
10	523	82.3	3

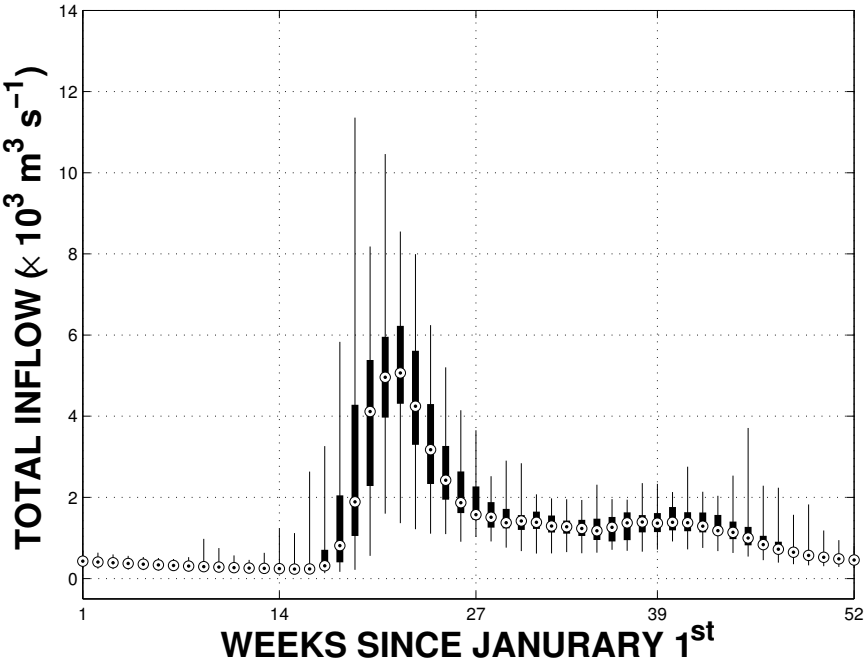

Figure 6.5 Historical inflow for the period 1962–2003.

The control vector

$$u_t := (q_{it}, D_{it}, P_{it}^{hydro}, P_t^{therm}, S_t)$$

at time period (week) t contains the turbined outflow q_{it} (m^3 s^{-1}) of hydro plants i during t, the spilled outflow D_{it} (m^3 s^{-1}) of hydro plant i during t, the power output P_{it}^{hydro} (MW) of hydro plant i during t, the power output P_t^{therm} (MW) of the thermal generating plant during t and the sales S_t (MW) time period t.

6.4.2 Energy balance

In this problem, the producer must satisfy the following energy balance constraints

$$\sum_{i=1}^{I} P_{it}^{hydro} + P_t^{therm} - S_t = L_t, \qquad \forall t = 1, ..., T$$

where L_t (MW) is the electrical load at time period t. For sake of simplicity, we assume that the load is a deterministic parameter in this problem. The total load is 69.5 TWh.

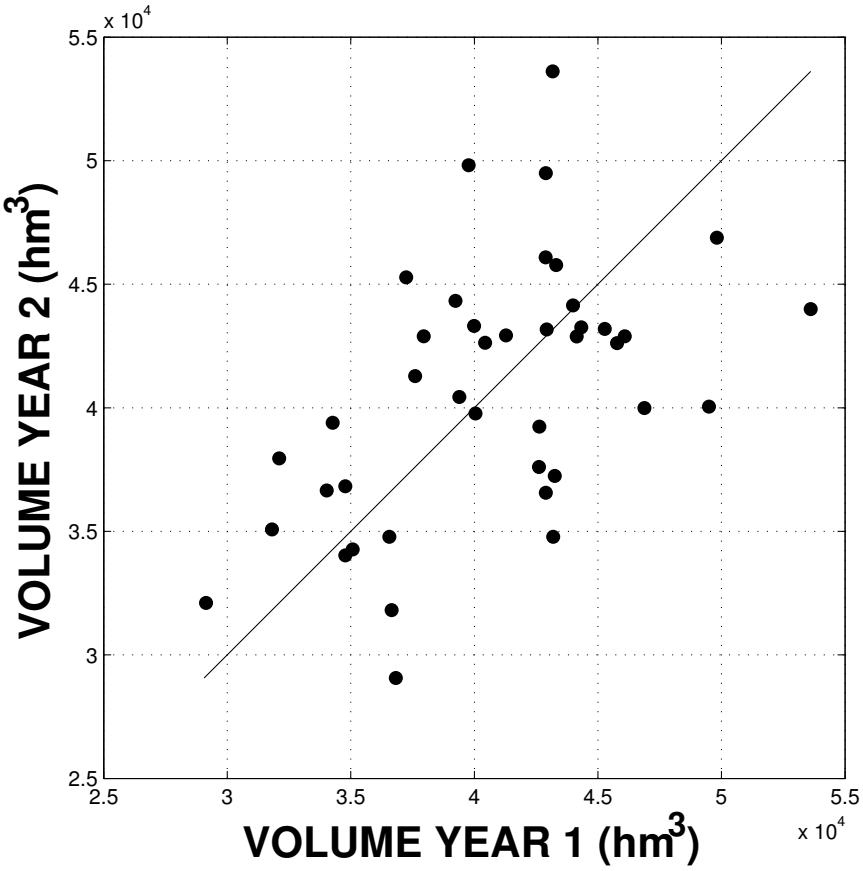

Figure 6.6 Scatter plot of correlation between each pair of consecutive years.

6.4.3 Hydroelectricity generation

We assume that the power output P_{it}^{hydro} of each hydro plant i is returned by a concave and piecewise linear function $\phi_i(q_{it}, v_{j(i)t})$ (MW) of the turbined outflow q_{it} and of the volume of water $v_{j(i)t}$ (hm^3) stored in the upstream reservoir $j(i)$. To use this approach, each production function

$$\phi_i(q_{it}, v_{j(i)t}) := \min_{h \in \mathcal{H}_i} \{\eta_{ih}^0 + \eta_{ih}^1 q_{it} + \eta_{ih}^2 v_{j(i)t}\}$$

is defined as the minimum value from a finite set of hyperplanes $h \in \mathcal{H}_i$. This type of approach is widely used in MOMs when large reservoirs are considered (Diniz et Maceira, 2008). Each hyperplane corresponds to a specific operating regime. The minimization operator is due to the concavity assumption. Fig. 6.8 shows an illustrative example of a unidimensional production function defined by three hyperplanes.

In the optimization model, the relationship between P_{it}, q_{it} and $v_{j(i)t}$ is represented by

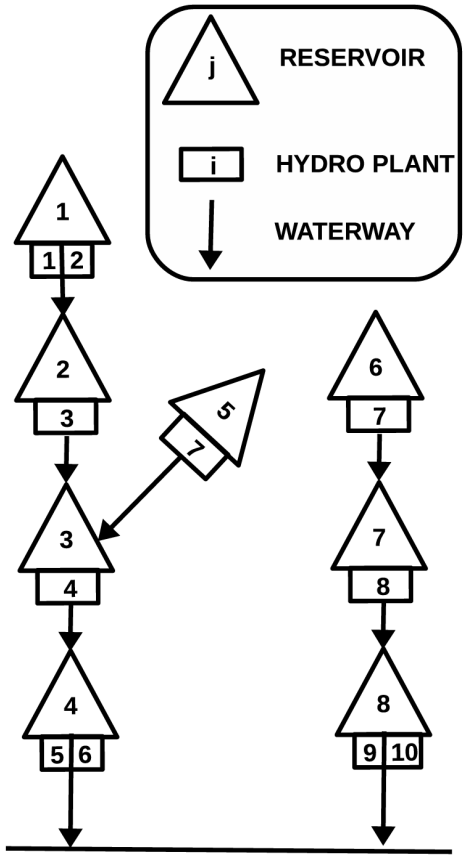

Figure 6.7 Power system.

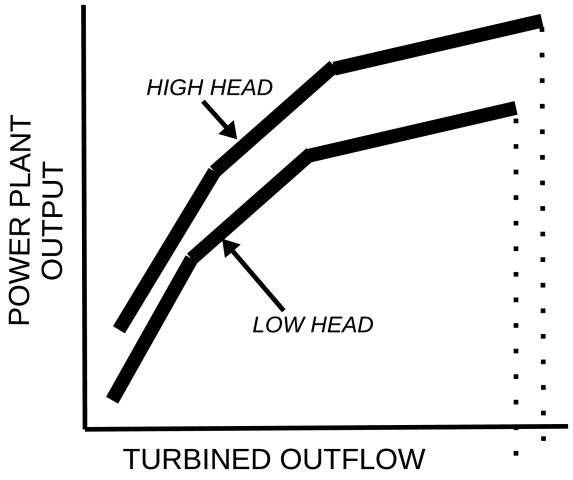

Figure 6.8 Hydroelectricity generation function.

the following inequality constraints

$$P_{it}^{hydro} \leq \eta_{ih}^{0} + \eta_{ih}^{1} q_{it} + \eta_{ih}^{2} v_{j(i)t}, \qquad \forall i, h, t$$

where η_{ih}^{0} (MW), η_{ih}^{1} (MW/m^{3}s^{-1}) and η_{ih}^{2} (MW/hm^3) are the coefficients of hyperplane h for plant i. In this experiment, each ϕ_i function was represented using three hyperplanes.

6.4.4 Water balance

The reservoir dynamics are governed by the following continuity equation

$$v_{jt} = v_{j,t-1} + \sum_{\gamma \in \Gamma(j)} Q_{\gamma t} - Q_{jt} + I_{jt}, \qquad \forall j, t$$

where I_{jt} (hm^3) is the volume of natural inflow in reservoir j during t and $\Gamma(j)$ is the set of reservoirs located immediately upstream of j in the hydrosystem. The volume of controlled outflow of reservoir j during t is defined as follows

$$Q_{jt} := \sum_{i \in \mathcal{I}(j)} (q_{it} + D_{it})\, \beta \qquad (\text{hm}^3)$$

where $\beta = 0.6048$ is constant for converting flow units into weekly volume and $\mathcal{I}(j)$ is the set of hydro plants supplied directly by reservoir j.

6.4.5 Bounds

The box constraints

$$\underline{v}_{jt} \leq v_{jt} \leq \overline{v}_{jt}^{\max} \quad , \qquad \forall j, t \tag{6.15}$$

$$0 \leq D_{it},\ 0 \leq q_{it} \leq \overline{q}_{it} \quad , \qquad \forall i, t \tag{6.16}$$

$$0 \leq P_{it}^{hydro} \leq \overline{P}_{it}^{hydro} \quad , \qquad \forall i, t \tag{6.17}$$

$$0 \leq P_{it}^{therm},\ S_t \quad , \qquad \forall t \tag{6.18}$$

represent the allowed operating range.

6.4.6 Objective function

Hydroelectricity generation is assumed to be cost free. We assume that that all fixed costs (start-up/shut-down, transaction) costs are negligible. The producer is assumed to be price-taker. Therefore, market price is not influenced by the production level of the controlled

power system. The net operating cost at time period t is defined as follows

$$g_t(u_t) := \left(\theta P_t^{therm} + \pi_t S_t\right) \Delta t$$

where $\theta = 60$ \$ MWh^{-1} is the marginal cost for thermal generation, π_t is a the selling price at time period t and $\Delta t = 168$ hours is the time step. We assume that the value of water in reservoirs is obtained as follows

$$\Psi(v_T) := \sum_{j=1}^{J} \psi_j v_{jT}$$

where ψ_j (\$ hm^{-3}) is a constant representing the marginal water value in reservoir j at the end of the horizon. In practice, these parameters are tuned using a long-term model. It corresponds to the marginal value of water at the end of the horizon.

6.4.7 Scenario tree

For sake of simplicity, we assume that the only random parameter in this problem is the volume of natural inflow I_{jt} (hm^3) in each reservoir j at t. All other parameters (load, prices) are assumed to be known with certainty over the entire planning horizon. We assume that the memory loss occurs at the end of week $\tau = 52$. We assume that $M = 1$ type of second-stage subtrees. We assume that first-stage subprocess $\{\xi_t : t = 1, ..., \tau\}$ and second-stage process $\{\xi_t : t = \tau + 1, ..., T\}$ have N_{br} different branches. Consequently, the full stochastic process $(\{\xi_t : t = 1, ..., T\})$ has $N_{scen} = N_{br}^2$ possible realizations (scenarios). Fig. 6.9 illustrated the procedure used to construct scenario trees. We used the backward scenario tree construction algorithm of the command line interface of SCENRED2 to create inflow subtrees with N_{br} branches from $N_{synth} = 10N_{br}$ synthetic time series. This software is an implementation of the method proposed by Heitsch et Römisch (2009) and is part of the General Algebraic Modeling System (GAMS) version 23.6.3. Each subtree was constructed from synthetically generated time series using a multivariate periodic autoregressive model of order 1 (MPAR(1)) using the Stochastic Analysis Modeling and Simulation (SAMS) software O. G. B. Sveinsson et Frevert (2007). Table 6.3 shows the characteristics of scenario trees that was constructed in this experiment. Fig. 6.10 show the branching structure for scenario trees F.

6.4.8 Experimental set-up

In this experiment, we solve the MTPP using the ELS algorithm. We used $\epsilon = 0.01$ as a stopping criterion. The model is implemented in object-oriented C++ and uses the barrier solver of the ILOG CPLEX/Concert technology library version 12.4. All computational results obtained in this study were obtained using personal computer running with a AMD

Tableau 6.3 Characteristics of scenario trees.

	SUBTREES				SCENARIO TREE	
	N_{br}	$\lvert\mathcal{N}_1\rvert$	$\lvert\mathcal{N}_2\rvert$	RF (%)	N_{scen}	$\lvert\mathcal{N}\rvert$
A	10	413	422	18.0	100	4,633
B	33	1,235	1,228	6.08	1,089	41,759
C	40	1,515	1,528	4.86	1,600	62,635
D	100	3,364	3,451	1.96	10,000	348,464
E	317	9,875	9,920	0.63	100,489	3,154,515
F	800	24,204	24,560	0.25	640,000	19,672,204

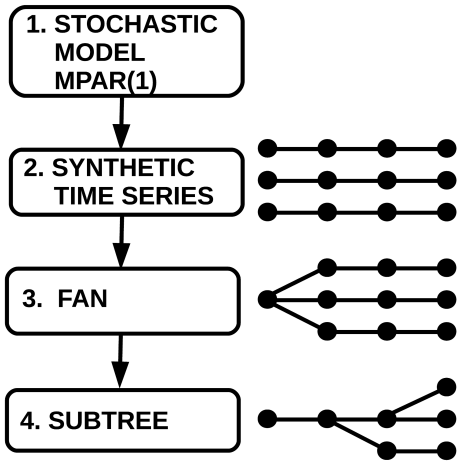

Figure 6.9 Procedure for constructing subtrees.

Phenom II X6 2.8 GHz processor and 6 GB of RAM. That computer runs on the 64 bits version of Ubuntu 12.04.

6.5 Results

Table 6.4 shows the algorithm's running time, the number of iterations, the random access memory (RAM) requirement, the expected sales (ES), the expected thermal generation (ETG) and the objective value z_* obtained for cases A–F. Table 6.5 shows the marginal water value (MWV) at the beginning of the horizon for each reservoir. The MWV_j (\$ hm^{-3}) for reservoir j is defined as the Lagrange multiplier associated with constraint (6.1). The ELS model can solve huge problems in reasonable running time. It is worth mentioning that the performance of our algorithm could be improved substantially by solving subproblems in

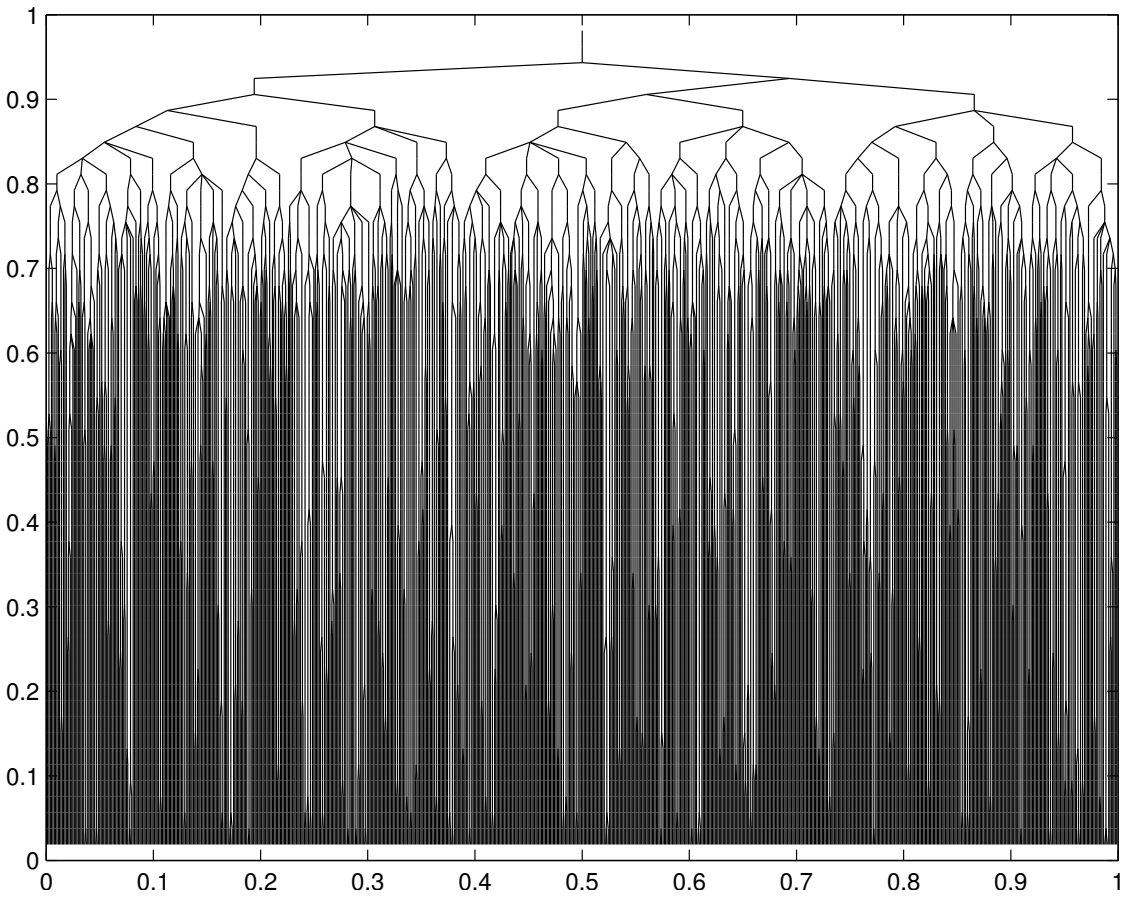

Figure 6.10 Branching structure of the first-stage subtree (case F).

parallel instead of sequentially. The largest scenario tree that we considered (case F) can be managed because we represent it implicitly with only two subtrees containing a total of 48,764 nodes. The equivalent complete scenario tree would contain a total of 19,672,204 nodes. Consequently, traditional methods such as the PH or the NBD could not be used to solve this problem. We notice that the problem's primal (thermal generation, sales) and dual solutions (MWV) vary depending on which scenario tree is used in the model.

6.6 Conclusions

In this paper, we considered the MTPP under uncertainty facing large hydroelectricity producers. We model uncertainty using a special class of STs. We partitioned the planning horizon in two consecutive stages where each stage contains many time periods (e.g. several months) and we assumed that the stochastic process driving the problem's random parameters has a memory loss at the end of the first stage. We demonstrated how the L-Shaped method

Tableau 6.4 Results obtained with the ELS algorithm.

	TIME (hours)	ITERATIONS	RAM (MB)	ES (GWh)	ETG (GWh)	z_* (M$)
A	0.02	6	60.8	784.2	2,553	-17.81
B	0.17	3	208	859.1	2,119	-40.41
C	0.31	4	255	842.9	3,840	50.01
D	1.52	3	556	861.0	3,937	70.44
E	9.48	3	1,381	932.0	3,898	24.51
F	41.1	2	3,356	920.0	3,940	34.04

Tableau 6.5 Marginal water value for each reservoir (\$ hm^{-3}).

j	A	B	C	D	E	F
1	49,961	49,197	49,716	49,665	49,201	49,283
2	29,372	28,962	29,272	29,235	29,011	29,029
3	15,433	15,218	15,381	15,361	15,243	15,253
4	5,034	4,963	5,016	5,010	4,972	4,975
5	40,160	39,600	40,023	39,973	39,651	39,660
6	47,094	46,639	47,715	47,879	47,773	47,803
7	31,881	31,639	32,555	32,738	32,748	32,768
8	10,770	10,822	11,515	11,724	11,895	11,903

can be extended to exploit the MLA. A numerical experiment was performed to show the computational benefit of using the proposed approximation. It enables to consider ST that would be too large to fit into most computer's hard disks. Therefore these STs conventional STBMs could not be used to solve the associated DEP.

CHAPITRE 7

DISCUSSION GÉNÉRALE

Les résultats présentés au chapitre 4 montrent que l'APH classique est applicable pour la gestion de grands systèmes hydroélectriques lorsque

- l'horizon de planification à moyen terme couvre plusieurs dizaines de périodes ;
- l'arbre de scénarios contient plusieurs étapes de branchement ;
- l'arbre de scénarios contient des scénarios hautement variables.

Cette méthode permet de calculer des cibles hebdomadaires non-anticipative en faisant un compromis entre les différents scénarios de l'arbre considéré. La solution optimale retournée par l'APH est plus facile à interpréter par les gestionnaires d'Hydro-Québec qu'une série de solutions déterministes complètement anticipatives calculées pour les mêmes scénarios. Les trois principales limitations de l'APH classique sont liées

1. au temps de calcul élevé requis pour satisfaire toutes les CNA (nombre d'itérations et temps de calcul par itération) ;
2. à la sensibilité du nombre d'itérations de l'algorithme au choix du paramètre de pénalité utilisé ;
3. au faible niveau de discrétisation de la distribution de probabilité continue caractérisant les paramètres aléatoires du problème.

La méthode de partitionnement optimal présentée au chapitre 5 s'attaque aux deux premières limitations de l'APH classique. Les résultats numériques obtenus montrent que notre méthode permet de réduire le temps de calcul de l'APH pour la résolution de programmes stochastiques multiétape. L'amélioration est expliquée par une accélération du taux de convergence de l'algorithme et par une réduction du temps de calcul par itération. Les résultats numériques obtenus montrent aussi que l'utilisation d'un schéma de décomposition par groupe de scénario permet de diminuer la sensibilité de l'APH au choix du paramètre de pénalité.

L'hypothèse de perte de mémoire combinée à la méthode L-Shaped étendue présentée au chapitre 5 s'attaque à la troisième limitations de l'APH classique. Notre approche permet d'augmenter le niveau de branchement de l'arbre de scénarios lorsque l'horizon de planification à moyen terme couvre un grand nombre de périodes. Les résultats obtenus montrent

que notre approche permet de considérer des arbres de scénarios qui sont hors de portée des méthodes traditionnelles de décomposition (p. ex. APH, nested Benders). L'approche proposée peut être combinée aux méthodes traditionnelles en les utilisant pour la résolution du problème maître ou du sous-problème.

CHAPITRE 8

CONCLUSIONS ET RECOMMANDATIONS

8.1 Synthèse des travaux

Dans cette thèse, nous avons développé et évalué différentes méthodes de décomposition permettant de résoudre efficacement le PPMT sous incertitude. Ces méthodes reposent sur une représentation par AS de l'incertitude et sont applicables aux systèmes multiréservoir de haute dimension exploités par les grands producteurs hydroélectriques. La formulation mathématique du PPMT que nous étudions est très générale et peut être adaptée pour représenter la majorité des situations pratiques. Cette formulation permet notamment de représenter différentes composantes additionnelles (p. ex. centrales thermiques, évacuateurs, marchés d'achat/vente, réseau de transport) et d'imposer une variété de contraintes opérationnelles (p. ex. demande électrique multiclasse, rampes...). Les paramètres aléatoires considérés dans cette formulation peuvent affecter l'objectif (prix) et les contraintes (p. ex. demande, apports, production intermittente). La méthode de partitionnement optimal présentée au chapitre 5 est applicable dans une contexte plus général que le PPMT. En effet, cette méthode peut être appliquée pour la résolution de PSM généraux.

Les trois articles présentés dans cette thèse contribuent de différentes façons à l'avancement des connaissances en gestion de grands systèmes hydroélectriques sur l'horizon à moyen terme. Au chapitre 4, nous démontrons l'applicabilité d'une méthode de décomposition par scénarios existante (l'APH) pour la gestion de réservoirs hydroélectriques multiannuels. Au chapitre 5, nous proposons une nouvelle approche permettant d'améliorer la performance computationnelle de cette méthode de décomposition pour la résolution de PSM généraux. Le chapitre 6 présente une nouvelle méthode de décomposition à deux étapes qui peut être combinée à une méthode de décomposition existante (p. ex. APH, nested Benders) pour la gestion de grands réservoirs hydroélectriques.

8.2 Limitations de la solution proposée

Dans l'ensemble, la qualité des solutions retournées par les méthodes de décomposition proposées dans cette thèse est limitée principalement par la qualité de l'arbre de scénarios utilisé pour représenter les paramètres aléatoires. Malgré la grande importance de cette problématique, nous n'avons pas exploré cette avenue pour des raisons de contraintes tempo-

relles. Nous avons plutôt choisi de concentrer nos efforts sur le développement de méthodes efficaces de résolution. Il n'en demeure par moins que la construction d'arbre est une tâche critique pour la planification à moyen terme de la production hydroélectrique. En raison de l'augmentation exponentielle de la taille de l'AS, les méthodes de décomposition basées sur un représentation par AS sont limitées à nombre restreint d'étapes de branchement et à un nombre modeste de branches par étape. Conséquemment, la topologie de l'AS utilisé pour modéliser l'incertitude doit être choisie soigneusement afin de maximiser la qualité des solutions retournées tout en respectant les contraintes informatiques (temps, mémoire) à satisfaire.

8.3 Améliorations futures

Une améliorations possible serait de proposer une nouvelle méthode pour la mise à jour du paramètre de pénalité. Aux chapitres 4 et 5, nous avons utilisé la formule traditionnelle des méthodes générales d'optimisation basées sur le Lagrangien augmenté. Nous avons aussi utilisé le même $\rho_k \in \mathbb{R}$ pour pénaliser toutes les CNA. Il est probable que l'utilisation d'une autre stratégie de mise à jour et que l'utilisation d'un paramètre ρ_k différent pour chaque contrainte accélère le taux de convergence de l'algorithme. La formule de mise à jour que nous avons considéré nécessite un calibrage des constantes ρ_0 et μ, ce qui peut être fastidieux en pratique. Il est possible que d'autres paramétrage de ρ_k simplifie le processus de calibration. Une autre amélioration possible de nos travaux serait d'étendre la méthode L-Shaped à deux étapes en un contexte multiétape. Autrement dit, le processus stochastique $\{\xi_t : t = 1, ..., T\}$ pourrait subir plusieurs pertes de mémoire τ_1, τ_2, ..., τ_M. Ceci permettrait d'augmenter significativement le niveau de branchement de l'arbre.

LISTE DES RÉFÉRENCES

ARCHIBALD, T. W., BUCHANAN, C. S., MCKINNON, K. I. M. et THOMAS, L. C. (1999). Nested benders decomposition and dynamic programming for reservoir optimization. *Journal of the Operational Research Society*, vol. 50, pp. 468–479.

ARCHIBALD, T. W., MCKINNON, K. I. M. et THOMAS, L. C. (1997). An aggregate stochastic dynamic programming model of multireservoir system. *Water Resources Research*, vol. 33, pp. 333–340.

BELLMAN, R. (1957). *Dynamic Programming*. Princeton University Press.

BENDERS, J. F. (1962). Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, vol. 4, pp. 238–252.

BERTSEKAS, D. P. et TSITSIKLIS, J. N. (1996). *Neuro-dynamic programming*. Athena Scientific.

BHASKAR, N. R. et WHITLATCH, E. E. (1980). Derivation of monthly reservoir release policies. *Water Resources Research*, vol. 16, pp. 987–993.

BIRGE, J. R. (1985). Decomposition and partitioning methods for multistage stochastic linear programs. *Operations Research*, vol. 33, pp. 989–1007.

BIRGE, J. R. et LOUVEAUX, F. (2011). *Introduction to Stochastic Programming*. Series in Operations Research. Springer, New York, seconde édition.

BIRGE, J. R. et LOUVEAUX, F. V. (1988). A multicut algorithm for two-stage stochastic linear programs. *European Journal on Operational Research*, vol. 34, pp. 384–392.

CARPENTIER, P.-L., GENDREAU, M. et BASTIN, F. (2013a). The extended l-shaped method for mid-term planning of hydroelectricity generation. *IEEE Transactions on Power Systems*, (submitted).

CARPENTIER, P.-L., GENDREAU, M. et BASTIN, F. (2013b). Long-term management of an hydroelectric multireservoir system under uncertainty using the progressive hedging algorithm. *Water Resources Research*, vol. 49, pp. 2812–2827.

CARPENTIER, P.-L., GENDREAU, M. et BASTIN, F. (2013c). Optimal scenario set partitioning for multistage stochastic programming with the progressive hedging algorithm. *European Journal of Operational Research*, (submitted).

CASTELLETTI, A., DE RIGO, D., RIZZOLI, A. E., SONCINI-SESSA, R. et WEBER, E. (2007). Neuro-dynamic programming for designing water reservoir network management policies. *Journal of Water Resources Planning and Management*, vol. 15, pp. 1031–1038.

CASTELLETTI, A., GALELLI, S., RESTELLI, M. et SONCINI-SESSA, R. (2010). Tree-based reinforcement learning for optimal water reservoir operation. *Water Resources Research*, vol. 46.

CONEJO, A. J., CASTILLO, E., MINGUEZ, R. et GARCIA-BERTRAND, R. (2006). *Decomposition Techniques in Mathematical Programming : Enginnering and Science Applications*. Springer.

CRAINIC, T. G., FU, X., GENDREAU, M., REI, W. et WALLACE, S. W. (2011). Progressive hedging-based metaheuristics for stochastic network design. *Networks*, vol. 58, pp. 114–124.

CRAINIC, T. G., HEWITT, M. et REI, W. (2013). Scenario grouping in a progressive hedging-based meta-heuristic for stochastic network design. *Computers & Operations Research*, pp. 1–25.

DINIZ, A. L. et MACEIRA, M. E. P. (2008). 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, pp. 1298–1308.

DOS SANTOS, T. N. et DINIZ, A. L. (2009). A new multiperiod stage definition for the multistage benders decomposition approach applied to hydrothermal scheduling. *IEEE Transactions on Power Systems*, vol. 4, pp. 1383–1392.

DUPAČOVÁ, J. (2002). Applications of stochastic programming : Achievements and questions. *European Journal of Operational Research*, vol. 140, pp. 281–290.

DUPAČOVÁ, J., CONSIGLI, G. et WALLACE, S. W. (2000). Scenarios for multistage stochastic programs. *Annals of Operations Research*, vol. 100, pp. 25–53.

FLETEN, S.-E. et KRISTOFFERSEN, T. K. (2008). Short-term hydropower production planning by stochastic programming,. *Computers and Operations Research*, vol. 35, pp. 2656–2671.

FORTIN, P. (2008). Canadian clean : Clean, renewable hydropower leads electricity generation in canada. *IEEE Power & Energy Magazine*, vol. 6, pp. 41–46.

GASSMAN, H. I. et ZIEMBA, W. T. (2013). *Stochastic Programming : Applications in Finance, Energy, Planning and Energy*. World Scientific.

GJELSVIK, A., MO, B. et HAUGSTAD, A. (2010). *Handbook on Power Systems I*, Springer, chapitre Long- and Medium-term Operations Planning and Stochastic Modelling in Hydro-dominated Power Systems Based on Stochastic Dual Dynamic Programming. Series on Energy Systems. pp. 33–55.

GONCALVES, R. E. C., FINARDI, E. C. et DA SILVA, E. L. (2011). Applying different decomposition schemes using the progressive hedging algorithm to the operation planning problem of a hydrothermal system. *Electric Power Systems Research*, vol. 137, pp. 258–267.

GOOR, Q., KELMAN, R. et TILMANT, A. (2011). Optimal multipurpose-multireservoir operation model with variable productivity of hydropower plants. *Journal of Water Resources Planning and Management*, vol. 59, pp. 258–267.

GRÖWE-KUSKA, N., HEITSCH, H. et RÖMISCH, W. (2003). Scenario reduction and scenario tree construction for power management problems. *IEEE PowerTech Conference*. Italy.

HAUGEN, K. K., LOKKETANGEN, A. et WOODRUFF, D. L. (2001). Progressive hedging as a meta-heuristic applied to stochastic lot-sizing. *European Journal of Operational Research*, vol. 132, pp. 116–122.

HEITSCH, H. et RÖMISCH, W. (2005). Generation of multivariate scenario trees to model stochasticity in power management. *IEEE PowerTech Conference*. Russia.

HEITSCH, H. et RÖMISCH, W. (2009). Scenario tree modeling for multistage stochastic programs. *Mathematical Programming Series A*, vol. 118, pp. 371–406.

HEITSCH, H., W. RÖMISCH, W. et STRUGAREK, C. (2006). Stability of multistage stochastic programs. *SIAM Journal on Optimization*, vol. 17, pp. 511–525.

HøYLAND, K. et WALLACE, S. W. (2001). Generating scenario trees for multistage decision problems. *Management Science*, vol. 47, pp. 295–307.

JACOBS, J., FREEMAN, G., GRYGIER, J., MORTON, D., SCHULTZ, G., STASCHUS, K. et STEDINGER, J. (1995). Socrates : A system for scheduling hydroelectric generation under uncertainty. *Annals of Operations Research*, vol. 59, pp. 99–133.

KALLRATH, J., PARDALOS, P. M., REBENNACK, S. et SCHEIDT, M. (2009). *Optimization in the Energy Industry*. series on Energy Systems. Springer-Verlag.

KARAMOUZ, M. et HOUCK, M. H. (1982). Annual and monthly reservoir operating rules generated by deterministic optimization. *Water Resources Research*, vol. 18, pp. 1337–1344.

KÜCHLER, C. et VIGERSKE, S. (2007). Decomposition of multistage stochastic programs with recombining scenario trees. *Stochastic Programming E-Print Series*, pp. 1–21.

LABADIE, J. (2004). Optimal operation of multireservoir systems : State-of-the-art review. *Journal of Water Resources Planning and Management*, vol. 130, pp. 93–111.

LAPORTE, G. et LOUVEAUX, F. V. (1993). The integer l-shaped method for stochastic integer programs with complete recourse. *Operations Research Letters*, vol. 13, pp. 133–142.

LATORRE, J. M., CERISOLA, S. et RAMOS, A. (2007). Clustering algorithms for scenario tree generation : Application to natural hydro inflows. *European Journal of Operational Research*, vol. 181, pp. 1339–1353.

LEE, J. H. et LABADIE, J. (2007). Stochastic optimization of multireservoir systems via reinforcement learning. *Water Resources Research*, vol. 43, pp. 1–16.

M. L. L. DOS SANTOS, E. C. FINARDI, E. L. D. S. et GONCALVES, R. E. C. (2009). Practical aspects in solving the medium-term operation planning problem of hydrothermal power systems by using the progressive hedging method. *International Journal of Electrical Power & Energy Systems*, vol. 137, pp. 258–267.

MACEIRA, M. E. P., DUARTE, V. S., PENNA, D. D. J., MORAES, L. A. M. et MELO, A. C. G. (2008). Ten years of application of stochastic dual dynamic programming in official and agent studies in brazil - description of the newave program. *16th Power Systems Computation Conference*. Glasgow, Scotland.

MULVEY, J. M. et VLADIMIROU, H. (1991). Applying the progressive hedging algorithm to stochastic generalized networks. *Annals of Operations Research*, vol. 31, pp. 399–424.

NANDALAL, K. D. W. et BOGARDI, J. J. (2007). *Dynamic Programming Based Operation of Reservoir : Applicability and Limits*. International Hydrology Series. Cambridge University Press, Cambridge, second edition édition.

NOCEDAL, J. et WRIGHT, S. (2006). *Numerical Optimization*. Series in Operations Research and Financial Engineering. Spri, New York, second edition édition.

O. G. B. SVEINSSON, J. D. SALAS, W. L. L. et FREVERT, D. K. (2007). Stochastic analysis modeling and simulation (sams) version 2007 : User's manual. Rapport technique Techinal Report no. 11,, Colorado State University.

PEREIRA, M. V. F. et PINTO, L. M. V. G. (1991). Multi-stage stochastic optimization applied to energy planning. *Mathematical Programming*, vol. 52, pp. 359–375.

PFLUG, G. C. (2001). Scenario tree generation for multiperiod financial optimization by optimal discretization. *Mathematical Programming Series B*, vol. 89, pp. 251–271.

PHILBRICK, J. R. et KITANIDIS, P. K. (1999). Limitations of deterministic optimization applied to reservoir operations. *Journal of Water Resources Planning and Management*, vol. 125, pp. 135–142.

POWELL, W. B. (2011). *Approximate Dynamic Programming : Solving the Curses of Dimensionality*. Wiley Series in Probability and Statistics. Wiley.

RANI, D. et MOREIRA, M. M. (2010). Simulation-optimization modeling : A survey and potential application in reservoir systems operation. *Water Resources Management*, vol. 24, pp. 1107–1138.

REBOURS, Y., KIRSCHEN, D., TROTIGNON, M. et ROSSIGNOL, S. (2007a). A survey of frequency and voltage control ancillary services - part i : Technical features. *IEEE Transactions on Power Systems*, vol. 22, pp. 350–357.

REBOURS, Y., KIRSCHEN, D., TROTIGNON, M. et ROSSIGNOL, S. (2007b). A survey of frequency and voltage control ancillary services - part ii : Economical features. *IEEE Transactions on Power Systems*, vol. 22, pp. 358–366.

ROCKAFELLAR, R. T. et WETS, R. J. B. (1991). Scenarios and policy aggregation in optimization under uncertainty. *Mathematics of Operations Research*, vol. 16, pp. 119–147.

ROTTING, R. et GJELSVIK, A. (1992). Stochastic dual dynamic programming for seasonal scheduling in the norwegian power system. *IEEE Transactions on Power Systems*, vol. 7, pp. 273–279.

RUSCZYNZKI, A. (2003). *Decomposition methods*, Elsevier, New York, chapitre Stochastic Programming in Transportation and Logistics. pp. 555–635.

RUSCZYNZKI, A. et SHAPIRO, A. (2003). *Stochastic Programming*. Elsevier.

SAGASTIZABAL, C. (2012). Divide to conquer : decomposition methods for energy optimization. *Mathematical Programming Series B*, vol. 134, pp. 187–222.

SHAWWASH, Z. K., SIU, T. K. et RUSSELL, S. O. D. (2000). The b. c. hydro short term hydro scheduling optimization model. *IEEE Transactions on Power Systems*, vol. 15, pp. 1125–1131.

TEJADA-GUIBERT, J. A., JOHNSON, S. A. et STEDINGER, J. R. (1995). The value of hydrologic information in stochastic dynamic programming models of a multireservoir system. *Water Resources Research*, vol. 31, pp. 2571–2579.

TILMANT, A. et KELMAN, R. (2007). A stochastic approach to analyze trade-offs and risks associated with large-scale water resources systems. *Water Resources Research*, vol. 43, pp. 1–11.

TILMANT, A., PINTE, D. et GOOR, Q. (2008). Assessing marginal water values in multipurpose multireservoir systems via stochastic programming. *Water Resources Research*, vol. 44, pp. 1–17.

TURGEON, A. (1998). An aggregation-diaggregation approach to long-term reservoir management. *Water Resources Research*, vol. 34, pp. 3585–3594.

TURGEON, A. (2007). Stochastic optimization of multireservoir operation : The optimal reservoir trajectory approach. *Water Resources Research*, vol. 43, pp. 1–10.

VAN SLYKE, R. M. et WETS, R. (1969). L-shaped linear programs with applications to optimal control and stochastic programming. *SIAM Journal on Applied Mathematics*, vol. 17, pp. 638–663.

WALLACE, S. W. et FLETEN, S.-E. (2003). *Stochastic Programming*, Elsevier, chapitre Stochastic Programming Models in Energy.

WATSON, J.-P. et WOODRUFF, D. L. (2011). Progressive hedging innovations for a class of stochastic mixed-integer resource allocation problems. *Computational Management Science*, vol. 8, pp. 355–370.

WURBS, R. (1993). Reservoir-system simulation and optimization models. *Journal of Water Resources Planning and Management*, vol. 119, 455–472.

YAKOWITZ, S. (1982). Dynamic programming applications in water resources. *Water Resources Research*, vol. 18, pp. 673–696.

YEH, W. W. G. (1985). Reservoir management and operations models : A state-of-the-art review. *Water Resources Research*, vol. 21, pp. 1797–1818.