



HAL
open science

Programmation linéaire mixte robuste; Application au dimensionnement d'un système hybride de production d'électricité.

Pierre-Louis Poirion

► To cite this version:

Pierre-Louis Poirion. Programmation linéaire mixte robuste; Application au dimensionnement d'un système hybride de production d'électricité.. Langage de programmation [cs.PL]. Conservatoire national des arts et metiers - CNAM, 2013. Français. ⟨NNT : 2013CNAM0948⟩. ⟨tel-01136659⟩

HAL Id: tel-01136659

<https://theses.hal.science/tel-01136659v1>

Submitted on 27 Mar 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization



**ÉCOLE DOCTORALE INFORMATIQUE, TÉLÉCOMMUNICATION ET
ÉLECTRONIQUE (PARIS)**

ÉQUIPE OC – LABORATOIRE CEDRIC

THÈSE présentée par :

Pierre-Louis POIRION

soutenue le : **17 décembre 2013**

pour obtenir le grade de : **Docteur du Conservatoire National des Arts et Métiers**

Discipline/ Spécialité : **Informatique**

**Programmation linéaire mixte robuste ;
application au dimensionnement d'un système
hybride de production d'électricité**

THÈSE dirigée par :

M. BILLIONNET Alain

Mme COSTA Marie-Christine

PR, ENSIIE / CEDRIC - Cnam

PR, ENSTA ParisTech / CEDRIC - Cnam

RAPPORTEURS :

M. ARTIGUES Christian

M. SOURD Francis

Directeur de recherche, LAAS CNRS

HDR, SNCF / DIRECTION SI2R / DIR I&R PRODUCTION

JURY :

M. POSS Michael

M. ROUPIN Frédéric

M. VANDERBECK François

MCF, Université de Technologie de Compiègne

PR, Université Paris-Nord

PR, Université Bordeaux 1

Remerciements

Je tiens tout d'abord à remercier ceux sans qui cette thèse n'aurait jamais vu le jour. Merci Marie-Christine et Alain pour votre encadrement, votre disponibilité, vos nombreux et précieux conseils et pour votre gentillesse.

Je remercie également Christian Artigues et Francis Sourd de m'avoir fait l'honneur d'être rapporteurs de cette thèse et pour l'intérêt qu'ils y ont porté. Merci également à Michael Poss, Frédéric Roupin et François Vanderbeck d'avoir accepté de faire partie de mon jury.

Je remercie aussi l'ensemble des membres de l'équipe Optimisation Combinatoire, Alain F., Eric, Alain B., Marie-Christine, Sourour, Amélie, Agnès, Christophe, Cédric, et les membres de l'UMA à l'ENSTA pour l'accueil, l'ambiance et l'aide fournie. Merci à Christophe M. pour le bon fonctionnement de l'informatique et merci à Maurice pour son aide en programmation.

Merci à Régis, Anthony, Thibaut et Jean-Noël, pour leur aide précieuse qu'ils m'ont fourni durant leurs stages.

Je tiens aussi à remercier tous ceux qui ont partagé mon bureau au CNAM et à l'ENSTA et qui m'ont supporté pendant trois ans. Merci Grégoire pour tes ""jeux de mots"" qui ont contribué à détendre l'atmosphère, tes madeleines qui permettaient de me tenir en vie entre les repas, et pour ta précieuse relecture de ma thèse. Merci à toi Sabine pour les échanges sur de nombreux domaines et pour tes ""bons plans"" Groupon. Merci Nicolas pour tes conseils de fin de thèse et la nourriture que tu ramenaes au bureau. Merci Anthony pour ton aide en proba. Je tiens également à remercier Maxence et Junyi.

Je souhaite aussi remercier Cédric H. pour les nombreux échanges scientifiques que nous avons eu et qui ont contribué à un chapitre de ma thèse.

Je tiens aussi à remercier ma famille pour son soutien. Je remercie en particulier ma mère qui a accepté de relire une bonne partie de ma thèse. Merci également à tous les amis qui ont été présents pendant ces trois années : Jérémie, Jiro, Nathanaël, Youki, Sandra, Aurélie et tous ceux que j'oublie.

Enfin, je souhaite aussi remercier ma femme, Emi, qui a eu à me supporter pendant tout ce temps. Merci pour sa patience, son amour, sa bonne humeur et sa compréhension.

Table des matières

Remerciements	3
1 Introduction	9
I Étude théorique des programmes linéaires robustes bi-niveaux avec variables de recours continus	13
2 État de l'art sur l'optimisation robuste	15
2.1 Les approches robustes générales	15
2.1.1 L'approche de Soyster	15
2.1.2 L'approche de Kouvelis et Yu	16
2.1.3 L'approche de Ben-Tal et Nemirovski	17
2.1.4 L'approche de Bertsimas et Sim	19
2.2 Les approches bi-niveaux	19
3 Résolution du problème robuste de base	23
3.1 Le problème robuste	24
3.1.1 modélisation du problème	24
3.1.2 Définition de l'ensemble \mathcal{D}	26
3.2 Résolution du problème de recours $R(x)$	27
3.3 L'algorithme de génération de contraintes	29
3.4 Quelques généralisations	33
3.4.1 Incertitude dans le membre gauche des contraintes	33
3.4.2 Généralisation à d'autres ensembles d'incertitudes	35
4 Résolution du problème robuste sans la propriété de recours complet	37
4.1 Résolution du problème	37
4.1.1 Modélisation	38
4.1.2 Résolution	41
4.2 Application à un problème de production	48

5	Étude d'un problème robuste généralisé à un ensemble d'incertitude quelconque	51
5.1	Présentation du problème	51
5.2	Minimisation d'une fonction convexe et résolution du problème de recours	52
5.3	Résolution du problème et résultats de complexité	59
6	Prise en compte des probabilités dans l'approche robuste	61
6.1	Choix du paramètre δ	61
6.1.1	Cas d'une loi uniforme	63
6.1.2	Cas d'une loi normale tronquée	64
6.2	Borne en probabilité sur la valeur du programme de recours	65
6.2.1	Définition et modélisation du problème	65
6.2.2	Calcul de \mathbb{P}	68
6.2.3	Calcul du volume d'une union d'hypercubes	71
II	Dimensionnement robuste d'un système hybride de production d'électricité	77
7	Présentation du problème	79
7.1	Introduction	79
7.2	Résolution du problème	81
7.2.1	Modélisation du problème déterministe	81
7.2.2	Comportement optimal de la batterie et du générateur	87
7.2.3	Complexité	89
7.3	Résolution du problème robuste	91
7.4	Résultats numériques	94
8	Résolution du problème de recours en temps polynomial	97
8.1	L'algorithme de programmation dynamique	97
8.2	Comparaison avec l'approche précédente	104
9	Généralisations et Améliorations	105
9.1	Ajout de contraintes sur le polyèdre d'incertitude	105
9.2	Construction d'instances, choix de données numériques	107
9.2.1	Sensibilité à la température	108
9.2.2	Échelles de temps	108
9.2.3	Modèles de ressource solaire et éolienne	109
9.2.4	Choix des composants et calcul des coûts	111
9.3	Résultats numériques	112
9.3.1	Pertinence d'agréger les variables entières δ_t par 24 heures	112
9.3.2	Impact de la contrainte glissante	113
10	Conclusion	117

Résumé

Dans cette thèse, nous nous intéressons à l'optimisation robuste. Plus précisément, nous nous intéresserons aux problèmes linéaires mixtes bi-niveaux, c'est à dire aux problèmes dans lesquels le processus de décision est divisé en deux parties : dans un premier temps, les valeurs optimales des variables dites "de décisions" seront calculées ; puis, une fois que l'incertitude sur les données est levée, nous calculerons les valeurs des variables dites "de recours". Dans cette thèse, nous nous limiterons au cas où les variables de deuxième étape, dites "de recours", sont continues.

Dans la première partie de cette thèse, nous nous concentrerons sur l'étude théorique de tels problèmes. Nous commencerons par résoudre un problème linéaire simplifié dans lequel l'incertitude porte seulement sur le membre droit des contraintes, et est modélisée par un polytope bien particulier. Nous supposerons en outre que le problème vérifie une propriété dite "de recours complet", qui assure que, quelles que soient les valeurs prises par les variables de décisions, si ces dernières sont admissibles, alors le problème admet toujours une solution réalisable, et ce, quelles que soient les valeurs prises par les paramètres incertains. Nous verrons alors une méthode permettant, à partir d'un programme robuste quelconque, de se ramener à un programme robuste équivalent dont le problème déterministe associé vérifie la propriété de recours complet. Avant de traiter le cas général, nous nous limiterons d'abord au cas où les variables de décisions sont entières. Nous testerons alors notre approche sur un problème de production. Ensuite, après avoir remarqué que l'approche développée dans les chapitres précédents ne se généralisait pas naturellement aux polytopes qui n'ont pas des points extrêmes 0-1, nous montrerons comment, en utilisant des propriétés de convexité du problème, résoudre le problème robuste dans le cas général. Nous en déduirons alors des résultats de complexité sur le problème de deuxième étape, et sur le problème robuste. Dans la suite de cette partie nous tenterons d'utiliser au mieux les informations probabilistes que l'on a sur les données aléatoires pour estimer la pertinence de notre ensemble d'incertitude.

Dans la deuxième partie de cette thèse, nous étudierons un problème de conception de parc hybride de production d'électricité. Plus précisément, nous chercherons à optimiser un parc de production électrique constitué d'éoliennes, de panneaux solaires, de batteries et d'un générateur à diesel, destiné à répondre à une demande locale d'énergie électrique. Il s'agit de déterminer le nombre d'éoliennes, de panneaux solaires et de batteries à installer afin de répondre à la demande pour un coût minimum. Cependant, les données du problème sont très aléatoires. En effet, l'énergie produite par une éolienne dépend de la force et de la direction du vent ; celle produite par un panneau solaire, de l'ensoleillement et la demande en électricité peut être liée à la température ou à d'autres paramètres extérieurs. Pour résoudre ce problème, nous commencerons par modéliser le problème déterministe en un programme linéaire mixte. Puis nous appliquerons directement l'approche de la première partie pour résoudre le problème robuste associé. Nous montrerons

ensuite que le problème de deuxième étape associé, peut se résoudre en temps polynomial en utilisant un algorithme de programmation dynamique. Enfin, nous donnerons quelques généralisations et améliorations pour notre problème.

mots-clés : Programmation linéaire, Optimisation robuste, Programmation dynamique, Énergie renouvelable

Abstract

Robust optimization is a recent approach to study problems with uncertain data that does not rely on a prerequisite precise probability model but on mild assumptions on the uncertainties involved in the problem. We studied a linear two-stage robust problem with mixed-integer first-stage variables and continuous second stage variables. We considered column wise uncertainty and focused on the case when the problem doesn't satisfy a "full recourse property" which cannot be always satisfied for real problems. We also studied the complexity of the robust problem which is NP-hard and proved that it is actually polynomial solvable when a parameter of the problem is fixed. We then applied this approach to study a stand-alone hybrid system composed of wind turbines, solar photovoltaic panels and batteries. The aim was to determine the optimal number of photovoltaic panels, wind turbines and batteries in order to serve a given demand while minimizing the total cost of investment and use. We also studied some properties of the second stage problem, in particular that the second stage problem can be solvable in polynomial time using dynamic programming.

key-words : Linear Programming, Robust Optimization, Dynamic Programming, Renewable Energy

Chapitre 1

Introduction

Lorsque l'on résout un problème d'optimisation, on suppose généralement que les données du problème sont fixées une bonne fois pour toute. Cependant, dans un environnement réel, connaître parfaitement les données du problème comme : les demandes des clients, la météo, les réactions des concurrents, les cours de la bourse, ..., supposerait de pouvoir prévoir le futur. Nous ne pouvons donc pas nous permettre de supposer que toutes les données sont connues une bonne fois pour toute. Généralement, pour prendre en compte l'incertitude des données, on suppose que celles-ci suivent une loi de probabilité dont la distribution est connue ; le problème peut alors être résolu en utilisant la programmation stochastique. On cherche alors à minimiser ou à maximiser la valeur moyenne d'une fonction, sous certaines contraintes, c'est à dire que l'on cherche à trouver l'optimum de notre problème, "en moyenne". Cette approche présente néanmoins deux problèmes majeurs : en pratique, il est difficile de connaître précisément la loi de probabilité suivie par les données incertaines ; en outre l'approche stochastique augmente de manière considérable la taille du problème traité. On est ainsi amené à résoudre des problèmes qui comportent un très grand nombre de variables et de contraintes et qu'il est difficile de traiter dans un temps de calcul raisonnable.

Dans cette thèse, nous nous intéressons à l'optimisation robuste, où le but n'est pas de minimiser l'espérance d'une fonction sous des contraintes, mais de minimiser la fonction objectif lorsque le "pire cas" se produit. En effet, l'approche robuste consiste à modéliser l'incertitude sur les données par un ensemble de scénarios, c'est à dire : un ensemble contenant un certain nombre de réalisations possibles des données incertaines. Le pire cas correspond alors, à variables fixées, au scénario de cet ensemble qui engendre le plus grand coût. Nous nous intéresserons plus particulièrement aux problèmes linéaires mixtes bi-niveaux, c'est à dire aux problèmes dans lesquels le processus de décision est divisé en deux parties : dans un premier temps, les valeurs optimales des variables dites "de décisions" seront calculées ; puis, une fois que l'incertitude sur les données est levée, nous calculerons les valeurs des variables dites "de recours". Dans cette thèse, nous nous limiterons

au cas où les variables de deuxième étape, dites "de recours", sont continues

Cette thèse est composée de deux parties. La première, sera consacrée à l'étude théorique de tels problèmes. Après avoir rappelé les principaux résultats qui ont été trouvés jusqu'à maintenant en optimisation robuste, nous étudierons au chapitre trois, un problème linéaire simplifié. Nous supposons que l'incertitude porte seulement sur le membre droit des contraintes, et nous nous limiterons au cas où l'ensemble des incertitudes est modélisé par un polytope bien particulier. Nous supposons en outre que le problème vérifie une propriété dite "de recours complet", qui assure que, quelles que soient les valeurs prises par les variables de décisions, si ces dernières sont admissibles, alors le problème admet toujours une solution réalisable, et ce, quelles que soient les valeurs prises par les paramètres incertains. Cette hypothèse est très forte et ne peut malheureusement pas toujours être vérifiée pour des problèmes réels. Nous verrons alors au chapitre quatre une méthode permettant, à partir d'un programme robuste quelconque, de se ramener à un programme robuste équivalent dont le problème déterministe associé vérifie la propriété de recours complet. Avant de traiter le cas général, nous nous limiterons d'abord au cas où les variables de décisions sont entières. Nous testerons ensuite notre approche sur un problème de production. Dans le chapitre cinq, nous supposons que l'ensemble d'incertitude est à présent modélisé par un polytope quelconque. Après avoir remarqué que l'approche développée dans les chapitres précédents ne se généralisait pas naturellement aux polytopes qui n'ont pas des points extrêmes $0 - 1$, nous montrerons comment, en utilisant des propriétés de convexité du problème, résoudre le problème robuste dans le cas général. Nous en déduisons alors des résultats de complexité sur le problème de deuxième étape, et sur le problème robuste. Dans le chapitre six, nous tenterons d'utiliser au mieux les informations probabilistes que l'on a sur les données aléatoires pour estimer la pertinence de notre ensemble d'incertitude, \mathcal{D} . En effet, supposons que cet ensemble ne corresponde pas à la réalité (s'il ne contient pas certains cas très probables par exemple), nous devons pouvoir estimer les chances d'obtenir une solution plus coûteuse que la solution robuste. En outre, lorsque \mathcal{D} est défini en fonction de paramètres, comment peut-on les fixer pour avoir le "meilleur" ensemble \mathcal{D} possible ? En effet, plus cet ensemble d'incertitude comprend de scénarios et plus la solution trouvée est robuste. Cependant nous voudrions éviter les solutions trop conservatrices qui ont un coût beaucoup plus élevé que celui d'une solution du problème déterministe. Nous tenterons donc de fixer au mieux ces paramètres afin d'obtenir une solution robuste peu chère mais suffisamment conservatrice.

L'épuisement des ressources traditionnelles a permis un regain d'intérêt pour l'utilisation des sources d'énergies renouvelables. Malheureusement, le caractère aléatoire et discontinu de ces énergies, dû à la difficulté de prévoir avec précision les variations météorologiques, rend leur production difficile à maîtriser.

Dans la deuxième partie de cette thèse, nous étudierons un problème de concep-

tion de parc hybride de production d'électricité. Plus précisément, nous chercherons à optimiser un parc de production électrique constitué d'éoliennes, de panneaux solaires, de batteries et d'un générateur à diesel, destiné à répondre à une demande locale d'énergie électrique. Il s'agit de déterminer le nombre d'éoliennes, de panneaux solaires et de batteries à installer afin de répondre à la demande pour un coût minimum. Cependant, les données du problème sont très aléatoires. En effet, l'énergie produite par une éolienne dépend de la force et de la direction du vent ; celle produite par un panneau solaire, de l'ensoleillement et la demande en électricité peut être liée à la température ou à d'autres paramètres extérieurs. Dans le chapitre huit, après avoir modélisé le problème déterministe en un programme linéaire mixte, et étudié sa complexité, nous appliquerons directement l'approche de la première partie pour résoudre le problème robuste associé. Dans le chapitre suivant, nous montrerons que le problème de deuxième étape associé, peut se résoudre en temps polynomial en utilisant un algorithme de programmation dynamique. Enfin, dans le chapitre dix, nous donnerons quelques généralisations et améliorations pour notre problème.

Première partie

Étude théorique des programmes linéaires robustes bi-niveaux avec variables de recours continues

Chapitre 2

État de l'art sur l'optimisation robuste

Dans ce chapitre, nous allons présenter quelques approches, dites robustes, qui permettent de traiter des programmes linéaires contenant des données incertaines. Ce chapitre sera divisé en deux parties. Dans un premier temps, nous présenterons des approches robustes dites "générales", puis dans un deuxième temps, nous nous concentrerons sur les approches bi-niveaux.

2.1 Les approches robustes générales

2.1.1 L'approche de Soyster

Le concept d'optimisation robuste a été introduit par Soyster en 1979 dans [34]. Son idée était de chercher une solution qui soit réalisable quel que soit le scénario considéré. Plus précisément, soit le programme linéaire mixte ci-dessous :

$$P \quad \begin{cases} \min_x c \cdot x \\ \sum_{j=1}^n a_{ij} x_j \leq b_i, \quad i = 1, \dots, m \\ x_i \in \mathbb{N}, \quad i = 1, \dots, n_1, \quad x_i \in \mathbb{R}_+, \quad i = (n_1 + 1), \dots, n \end{cases}$$

Supposons que l'incertitude porte uniquement sur les coefficients a_{ij} de la matrice $A = (a_{ij})_{(i,j) \in [1, \dots, m] \times [1, \dots, n]}$ de taille $m \times n$.

L'ensemble d'incertitude adopté pour modéliser les coefficients de la matrice A est un modèle par intervalles, que nous définissons de la manière suivante : pour tout $i = 1, \dots, m$, et pour tout $j = 1, \dots, n$, le paramètre a_{ij} appartient à l'intervalle $[\bar{a}_{ij} - \hat{a}_{ij}, \bar{a}_{ij} + \hat{a}_{ij}]$, où \bar{a}_{ij} et \hat{a}_{ij} sont donnés. En d'autres termes, il existe des variables $\zeta_{ij} \in [-1, 1]$ pour tout i, j telles que

$$a_{ij} = \bar{a}_{ij} + \hat{a}_{ij} \zeta_{ij}.$$

L'approche proposé par Soyster, consiste alors à résoudre le problème suivant :

$$P_{Soy} \left| \begin{array}{l} \min_x c \cdot x \\ \sum_{j=1}^n (\bar{a}_{ij} + \hat{a}_{ij} \zeta_{ij}) x_j \leq b_i, \forall \zeta_{ij} \in [-1, 1], i = 1, \dots, m \\ x_i \in \mathbb{N}, i = 1, \dots, n_1, x_i \in \mathbb{R}_+, i = (n_1 + 1), \dots, n \end{array} \right.$$

qui revient à résoudre

$$P_{Soy} \left| \begin{array}{l} \min_x c \cdot x \\ \sum_{j=1}^n (\bar{a}_{ij} + \hat{a}_{ij}) x_j \leq b_i, i = 1, \dots, m \\ x_i \in \mathbb{N}, i = 1, \dots, n_1, x_i \in \mathbb{R}_+, i = (n_1 + 1), \dots, n \end{array} \right.$$

Cette approche permet donc de se ramener à un programme linéaire mixte de même nature que le problème déterministe initial. Cependant, en cherchant une solution qui soit valide quelles que soient les valeurs des données incertaines, nous nous limitons à des solutions trop conservatrices, dont la valeur est trop éloignée de celle d'une solution optimale du programme déterministe initial.

2.1.2 L'approche de Kouvelis et Yu

En 1997, Kouvelis et Yu proposent dans [27] une nouvelle modélisation de l'incertitude. Ils supposent que les données incertaines du problème appartiennent, non plus à un ensemble convexe, mais à un ensemble discret dont les éléments sont assimilés à des "scénarios". Plus précisément, soit le problème d'optimisation suivant :

$$P \left| \begin{array}{l} \min_x f(x) \\ x \in X \end{array} \right.$$

où X est l'ensemble des contraintes du problème et où f est la fonction de coût. Les données étant incertaines, nous supposons que nous avons à notre disposition un ensemble \mathcal{S} de scénarios qui modélisent l'incertitude des données. Nous écrivons alors que pour le scénario $s \in \mathcal{S}$, la fonction de coût f devient f_s , et l'ensemble des incertitudes X devient X_s .

Comme dans l'approche de Soyster, nous cherchons une solution qui soit admissible quel que soit le scénario $s \in \mathcal{S}$. Ainsi nous cherchons un vecteur x tel que :

$$x \in \bigcap_{s \in \mathcal{S}} X_s.$$

Kouvelis et Yu proposent trois critères pour définir la fonction coût.

Le premier, appelé "critère du pire cas", consiste à chercher une solution qui minimise le coût engendré par le pire scénario associé à cette solution. Nous cherchons alors à résoudre :

$$P_{PC} \left| \begin{array}{l} \min_x \max_s f_s(x) \\ x \in \bigcap_{s \in \mathcal{S}} X_s \end{array} \right.$$

Dans les deux autres approches, nous cherchons cette fois-ci à minimiser le plus grand écart entre le coût de notre solution et le coût d'une solution optimale lorsqu'on connaît le scénario s à prendre en compte. Formellement, soit x_s^* une solution optimale du problème déterministe dans le cas où le scénario considéré est s , le critère du pire "regret absolu" s'écrit :

$$P_{PRA} \left| \begin{array}{l} \min_x \max_s f_s(x) - f_s(x_s^*) \\ x \in \bigcap_{s \in \mathcal{S}} X_s \end{array} \right.$$

De même, le critère du pire "regret en proportion" demande de résoudre :

$$P_{PRA} \left| \begin{array}{l} \min_x \max_s \frac{f_s(x) - f_s(x_s^*)}{f_s(x_s^*)} \\ x \in \bigcap_{s \in \mathcal{S}} X_s \end{array} \right.$$

2.1.3 L'approche de Ben-Tal et Nemirovski

En cherchant une solution qui reste admissible quelles que soient les valeurs prises par les données incertaines, nous avons vu que les solutions engendrées par l'approche de Soyster avaient un coût largement supérieur à celui des solutions du problème déterministe. Pour remédier à ce problème, Ben-Tal et Nemirovski proposent, dans [11], une approche dans laquelle le domaine d'incertitude considéré exclut les valeurs extrêmes des intervalles en autorisant une faible violation des contraintes.

Considérons le programme linéaire mixte suivant :

$$P \left| \begin{array}{l} \max_x c \cdot x \\ \sum_{j=1}^n a_{ij} x_j \leq b_i, \quad i = 1, \dots, m \\ x_i \in \mathbb{N}, \quad i = 1, \dots, n_1, \quad x_i \in \mathbb{R}_+, \quad i = (n_1 + 1), \dots, n \end{array} \right.$$

et rappelons que les coefficients incertains a_{ij} , $i = 1, \dots, m$, $j = 1, \dots, n$ peuvent s'écrire

$$a_{ij} = \bar{a}_{ij} + \hat{a}_{ij} \zeta_{ij},$$

où les ζ_{ij} sont des variables qui appartiennent à $[-1, 1]$ pour tout i, j . Pour éviter les cas extrêmes les plus coûteux, les auteurs imposent, pour i , $i = 1, \dots, m$, pour

toute contrainte i , $i = 1, \dots, m$ du programme linéaire incertain, que les déviations maximales sur une même contrainte i appartiennent à un ellipsoïde défini par :

$$\Xi(\Omega_i) = \left\{ \zeta_i \in \mathbb{R}^n \mid \sqrt{\sum_{j=1}^n \zeta_{ij}^2} \leq \Omega_i, \zeta_{ij} \in [-1, 1] \right\}$$

où $\Omega_i \geq 0$, pour tout $i = 1, \dots, m$.

Le problème robuste considéré par Ben-Tal et Nemirovski s'écrit alors :

$$P_{BenTal} \left| \begin{array}{l} \max_x c.x \\ \max_{\zeta_i \in \Xi(\Omega_i)} \left\{ \sum_{i=1}^n \bar{a}_{ij}x_j + \hat{a}_{ij}\zeta_{ij}x_j \right\} \leq b_i, \quad i = 1, \dots, m \\ x_i \in \mathbb{N}, \quad i = 1, \dots, n_1, \quad x_i \in \mathbb{R}_+, \quad i = (n_1 + 1), \dots, n \end{array} \right.$$

qui est équivalent à

$$P_{BenTal} \left| \begin{array}{l} \max_x c.x \\ \sum_{j=1}^n \bar{a}_{ij}x_j + \max_{\zeta_i \in \Xi(\Omega_i)} \left\{ \sum_{i=1}^n \hat{a}_{ij}\zeta_{ij}x_j \right\} \leq b_i, \quad i = 1, \dots, m \\ x_i \in \mathbb{N}, \quad i = 1, \dots, n_1, \quad x_i \in \mathbb{R}_+, \quad i = (n_1 + 1), \dots, n \end{array} \right.$$

Cette approche consiste à considérer, pour chaque contrainte i , le pire scénario ζ_i possible. Ainsi ce modèle d'incertitude est souvent nommé dans la littérature par "modèle en ligne", du fait qu'un paramètre Ω_i soit défini pour chaque contrainte i .

Babonneau et al. ont montré dans [7], que le problème robuste ci-dessus est équivalent au programme quadratique suivant :

$$P_{BenTal} \left| \begin{array}{l} \max_{x,z} c.x \\ \sum_{j=1}^n \bar{a}_{ij}x_j + \Omega_i z_i \leq b_i, \quad i = 1, \dots, m \\ \sum_{i=1}^n \hat{a}_{ij}x_j^2 \leq z_i^2, \quad i = 1, \dots, m \\ x_i \in \mathbb{N}, \quad i = 1, \dots, n_1, \quad x_i \in \mathbb{R}_+, \quad i = (n_1 + 1), \dots, n, \quad z \in \mathbb{R}_+^m \end{array} \right.$$

En supposant que les variables aléatoires ζ_{ij} sont uniformément distribuées dans $[-1, 1]$, Ben-Tal et al. ont montré dans [10] que la probabilité de violation de la contrainte i ne dépasse pas $\exp\left(\frac{-\Omega_i^2}{2}\right)$.

2.1.4 L'approche de Bertsimas et Sim

L'approche que Bertsimas et Sim présentent dans [13] est similaire à l'approche de Ben-Tal et Nemirovski. La différence entre les deux approches est uniquement l'ensemble d'incertitude de ζ_i . En effet, nous supposons cette fois que le vecteur ζ_i appartient non plus à un ellipsoïde mais à l'ensemble :

$$\mathcal{D}_{\Gamma_i} = \left\{ \zeta_i \in \mathbb{R}^n \mid \sum_{j=1}^n |\zeta_{ij}| \leq \Gamma_i, \zeta_{ij} \in [-1, 1] \right\}.$$

Le problème robuste s'écrit alors :

$$P_{Bertsimas} \left\{ \begin{array}{l} \max_x c \cdot x \\ \sum_{j=1}^n \bar{a}_{ij} x_j + \max_{\zeta_i \in \mathcal{D}_{\Gamma_i}} \left\{ \sum_{i=1}^n \hat{a}_{ij} \zeta_{ij} x_j \right\} \leq b_i, \quad i = 1, \dots, m \\ x_i \in \mathbb{N}, \quad i = 1, \dots, n_1, \quad x_i \in \mathbb{R}_+, \quad i = (n_1 + 1), \dots, n \end{array} \right.$$

Bertsimas et Sim montrent que leur problème robuste est équivalent au programme linéaire suivant

$$P_{Bertsimas} \left\{ \begin{array}{l} \max_{x,y,t,p} c \cdot x \\ \sum_{j=1}^n \bar{a}_{ij} x_j + t_i \Gamma_i + \sum_{j=1}^n p_{ij} \leq b_i, \quad i = 1, \dots, m \\ t_i + p_{ij} \geq (\bar{a}_{ij} + \hat{a}_{ij}) \quad i = 1, \dots, m, \quad j = 1, \dots, n \\ -y_j \leq x_j \leq y_j, \quad j = 1, \dots, n \\ p_{ij} \geq 0 \quad i = 1, \dots, m, \quad j = 1, \dots, n \\ t_i, y_j \geq 0 \quad i = 1, \dots, m, \quad j = 1, \dots, n \\ x_i \in \mathbb{N}, \quad i = 1, \dots, n_1, \quad x_i \in \mathbb{R}_+, \quad i = (n_1 + 1), \dots, n \end{array} \right.$$

En outre, les auteurs montrent que si les ζ_{ij} sont des variables aléatoires symétriques de même loi sur $[-1, 1]$, alors la probabilité que la i ème contrainte soit violée est majorée par $\exp\left(\frac{-\Gamma_i^2}{2n}\right)$

2.2 Les approches bi-niveaux

Les approches robustes bi-niveaux sont fortement inspirées de l'optimisation stochastique. En effet, en optimisation stochastique, les incertitudes sont décrites par des lois de probabilités et les variables sont divisées en deux. Les variables de première étape (ou variables de décisions) doivent être déterminées avant que l'incertitude ne soit levée ; les variables de deuxième étapes (ou variables de recours) quant à elles, sont déterminées après que l'incertitude soit levée. L'objectif est de

En supposant que la propriété de recours complet soit vérifiée, et que le problème déterministe est borné pour tout d , Thiel et Minoux montrent que sous certaines conditions, il est possible de transformer le problème de recours en un problème linéaire mixte à variables 0 – 1. Le problème robuste est alors résolu à l’aide d’un algorithme de génération de colonnes. Notons que Minoux a prouvé dans [29] que le problème de recours $R(x)$ appartient à la classe des problèmes NP-difficiles.

Cette approche où l’incertitude est en colonne est utilisée dans de nombreuses applications, notamment dans [18], où Gabrel et al., considèrent un problème de localisation et de transport.

Chapitre 3

Résolution du problème robuste de base

Dans tout ce chapitre, nous nous intéresserons seulement aux problèmes vérifiant la propriété dite de recours complet. Nous définirons le problème robuste que nous allons étudier, ainsi que plusieurs sous-problèmes associés. Avant de considérer un problème plus restreint, nous donnerons quelques propriétés sur le problème robuste. Ensuite, par souci de clarté, nous supposerons dans un premier temps que seul le membre droit des contraintes comporte des coefficients incertains, et que l'ensemble d'incertitude sera celui introduit par Bertsimas et Sim dans [13] : tous les coefficients du membre droit des contraintes appartiennent à un intervalle qui modélise l'incertitude sur ce coefficient ; en outre la somme totale des variations est bornée. Afin de résoudre le problème robuste, nous commencerons par étudier le problème, dit "de recours", associé au problème robuste, qui permet de calculer le "pire scénario" lorsque les valeurs des variables de décision sont fixées. Nous verrons qu'il est possible de reformuler ce dernier en un programme linéaire mixte en nombres entiers. Afin de résoudre problème robuste, nous le reformulerons en un programme linéaire mixte, puis nous utiliserons un algorithme de génération de contraintes pour le résoudre. Enfin nous verrons que nos résultats s'étendent assez naturellement au cas où il y a aussi de l'incertitude dans le membre de gauche des contraintes et lorsque l'ensemble d'incertitude est défini de manière plus générale.

3.1 Le problème robuste

3.1.1 modélisation du problème

Soit le programme linéaire mixte suivant :

$$P \left\{ \begin{array}{l} \min_{x,y} \alpha \cdot x + \beta \cdot y \\ Ax + By \geq d \\ Cx \geq b \\ x_i \in \mathbb{N}, i = 1, \dots, p_1, x_i \in \mathbb{R}_+, i = (p_1 + 1), \dots, p, y \in \mathbb{R}_+^q \end{array} \right. \quad (3.1)$$

$$(3.2)$$

$$(3.3)$$

où $A \in \mathbb{Q}^{T \times p}$, $B \in \mathbb{Q}^{T \times q}$, $d \in \mathbb{Q}^T$, $C \in \mathbb{Q}^{n \times p}$, $b \in \mathbb{Q}^n$, $\alpha \in \mathbb{Q}^p$, $\beta \in \mathbb{Q}^q$, et où \mathbb{Q} est l'ensemble des nombres rationnels.

Définition 1. (x, y) est une solution admissible de P si et seulement si (x, y) vérifie toutes les contraintes de P . Dans ce cas on dit que P admet une solution

Rappelons que les variables x , qui sont entières ou continues, concernent les décisions de première étape et doivent être déterminées avant que l'incertitude ne soit levée ; les variables de seconde étape, y , appelées variables de recours, sont continues et sont déterminées après connaissance des coefficients incertains.

Nous supposons qu'il existe (x, y) tel que (3.1)-(3.3) sont vérifiées et nous dirons qu'une solution x est réalisable si x satisfait les contraintes (3.2) et (3.3). Dans cette section, les coefficients incertains ne sont que dans le membre droit des contraintes, d .

Pour un programme mathématique π , nous noterons par $v(\pi)$ la valeur d'une solution optimale du programme. Nous définirons la propriété \mathcal{P} , dite "propriété de recours complet", vérifiée par le programme P , qui assure que pour toute solution de première étape réalisable (ici x) et pour toute réalisation de d , il existe une valeur des variables de recours (ici y) telle que (3.1) soit satisfaite, c'est à dire : pour tout x vérifiant $Cx \geq d$, $x_i \in \mathbb{N}$, $i = 1, \dots, p_1$, $x_i \in \mathbb{R}_+$, $i = (p_1 + 1), \dots, p$ et pour tout $d \in \mathcal{D}$, il existe y vérifiant $By \geq d - Ax$, $y \in \mathbb{R}_+^q$. Remarquons que la propriété \mathcal{P} est toujours vérifiée si il existe une colonne de la matrice B dont tous les termes sont strictement positifs. Nous supposons que d appartient à un ensemble donné \mathcal{D} , défini par un polytope qui représente l'incertitude de d . Ce polytope, supposé quelconque pour l'instant, sera défini plus précisément à la fin de cette section. Pour tout $d \in \mathcal{D}$, nous parlerons de scénario possible pour le second membre de P .

Dans tout ce chapitre, nous supposerons que le problème déterministe vérifie la propriété \mathcal{P} . Nous supposerons en outre qu'il est borné inférieurement. Ainsi toute

Puisque P vérifie \mathcal{P} , $R(x, d)$ admet toujours une solution admissible. Nous allons maintenant montrer que pour tout x réalisable, le scénario extrême d^* associé à x est un point extrême du polytope \mathcal{D} .

Proposition 1. *Soit x réalisable, la fonction f de \mathcal{D} dans \mathbb{R} qui à tout scénario d associe $f(x, d) = v(R(x, d))$ est convexe.*

Démonstration. Soit $d_1, d_2 \in \mathcal{D}$ et soit y_1 et y_2 les vecteurs réels qui sont des solutions optimales de $R(x, d_1)$ et $R(x, d_2)$. On a, par définition de f :

$$f(x, d_1) = \beta \cdot y_1 \text{ et } f(x, d_2) = \beta \cdot y_2.$$

Soit $\lambda \in [0, 1]$, puisque \mathcal{D} est un polytope, $\lambda d_1 + (1 - \lambda)d_2 \in \mathcal{D}$, et $B(\lambda y_1 + (1 - \lambda)y_2) \geq \lambda(d_1 - Ax) + (1 - \lambda)(d_2 - Ax) = \lambda d_1 + (1 - \lambda)d_2 - Ax$. Donc $\lambda y_1 + (1 - \lambda)y_2$ est une solution admissible de $R(x, \lambda d_1 + (1 - \lambda)d_2)$, et par conséquent :

$$f(x, \lambda d_1 + (1 - \lambda)d_2) \leq \beta \cdot (\lambda y_1 + (1 - \lambda)y_2) = \lambda f(x, d_1) + (1 - \lambda)f(x, d_2).$$

□

Lemme 1. *Pour tout x réalisable, le scénario extrême d^* associé à x est un point extrême du polytope \mathcal{D} .*

Démonstration. Par définition, nous avons $v(R(x)) = \max_{d \in \mathcal{D}} f(x, d)$. Or puisque f est convexe, le maximum de f est atteint en un point extrême de \mathcal{D} . □

3.1.2 Définition de l'ensemble \mathcal{D}

Définissons maintenant plus précisément l'ensemble d'incertitude \mathcal{D} . Suivant l'idée proposée par Bertsimas et Sim [13] et Minoux [29], nous supposons que chaque coefficient d_t , $t = 1, \dots, T$ appartient à un intervalle $[\bar{d}_t - \Delta_t, \bar{d}_t + \Delta_t]$ où \bar{d}_t est donné et où $\Delta_t \geq 0$ est une borne donnée sur l'incertitude de d_t . L'ensemble d'incertitude \mathcal{D} est alors défini par :

$$\mathcal{D} = \{d : d_t \in [\bar{d}_t - \Delta_t, \bar{d}_t + \Delta_t], \forall t = 1, \dots, T\}.$$

Lorsque x est fixé, le pire scénario associé est obtenu pour $d_t = \bar{d}_t + \Delta_t$, $t = 1, \dots, T$. En effet, $By \geq \bar{d} + \Delta - Ax$ implique que $By \geq d - Ax$ pour tout $d \in \mathcal{D}$. Ainsi cette définition de l'incertitude ramène le problème robuste à un problème déterministe qui fournit une grande "protection" contre les incertitudes. Cependant cette définition de l'incertitude est trop prudente en pratique et implique des solutions avec des coûts très élevés. Pour éviter de trop contraindre le système nous ajoutons comme dans [35], la contrainte :

$$\sum_{\substack{t=1 \\ \Delta_t > 0}}^T \left| \frac{d_t - \bar{d}_t}{\Delta_t} \right| \leq \bar{\delta}, \quad (3.4)$$

où $\bar{\delta}$ est un entier positif qui borne la somme totale des variations des d_t . Soit, pour tout $t = 1, \dots, T$, la variable $\delta_t \in [-1, 1]$ modélisant l'incertitude sur d_t . On a alors $d_t = \bar{d}_t + \delta_t \Delta_t$, $\forall t = 1, \dots, T$, et l'ensemble d'incertitude \mathfrak{D} se réécrit :

$$\mathfrak{D} = \{d : d_t = \bar{d}_t + \delta_t \Delta_t, \sum_{t=1}^T |\delta_t| \leq \bar{\delta}, -1 \leq \delta_t \leq 1, \forall t = 1, \dots, T\}.$$

Remarquons qu'il existe toujours un scénario extrême d tel que $d_t \geq \bar{d}_t$ pour tout t . En effet, remplacer $\delta_t < 0$ par $-\delta_t > 0$ ne change pas la valeur de $\sum_{t=1}^T |\delta_t|$, mais augmente le coût induit par ce scénario. Nous pouvons donc nous restreindre sans perte de généralité au cas où $d_t \geq \bar{d}_t$. Dans tout le reste de cette section nous considérerons donc l'ensemble d'incertitude \mathfrak{D} défini par

$$\mathfrak{D} = \{d : d_t = \bar{d}_t + \delta_t \Delta_t, \sum_{t=1}^T \delta_t \leq \bar{\delta}, 0 \leq \delta_t \leq 1, \forall t = 1, \dots, T\}.$$

Remarquons que d_t atteint sa pire valeur $\bar{d}_t + \Delta_t$, pour au plus $\bar{\delta}$ valeurs de t , $0 \leq \bar{\delta} \leq T$. Les cas $\bar{\delta} = 0$ et $\bar{\delta} = T$ correspondent chacun à un seul scénario possible. $d_t = \bar{d}_t$, $t = 1, \dots, T$ pour $\bar{\delta} = 0$, $d_t = \bar{d}_t + \Delta_t$, $t = 1, \dots, T$, pour $\bar{\delta} = T$ (cas du pire scénario). Les autres valeurs de $\bar{\delta}$ correspondent chacune à un ensemble de scénarios.

3.2 Résolution du problème de recours $R(x)$

Pour résoudre le problème de recours (de type "max min") pour x fixé, le sous problème de minimisation est transformé en un problème de maximisation en considérant son dual. On obtient alors un problème de maximisation d'une fonction quadratique non concave. Nous montrons que, quels que soient les coefficients du problème de recours, il est toujours possible de réécrire les termes quadratiques en un produit d'une variable 0-1 et d'une variable continue et bornée. Nous pouvons alors linéariser ces termes quadratiques pour aboutir à un problème linéaire mixte.

P satisfaisant la propriété de recours complet, nous savons que pour tout x réalisable et pour tout $d \in \mathfrak{D}$, $R(x, d)$ admet une solution réalisable. En outre P étant borné inférieurement, nous en déduisons que $R(x, d)$ l'est aussi. Nous savons alors, par le théorème fort de la dualité :

$$v(R(x, d)) = v(DR(x, d)),$$

où $DR(x, d)$ est le programme dual de $R(x, d)$, c'est à dire

$$DR(x, d) \left\{ \begin{array}{l} \max_{\lambda} (d - Ax) \cdot \lambda \\ B^{tr} \lambda \leq \beta \\ \lambda \in \mathbb{R}_+^T. \end{array} \right. \quad (3.5)$$

Nous avons donc, pour tout x réalisable, $v(R(x)) = \max_{d \in \mathfrak{D}} v(DR(x, d))$.

Nous pouvons alors reformuler $R(x)$ en :

$$DR(x) \left| \begin{array}{l} \max_{\substack{\delta: \sum_{t=1}^T \delta_t \leq \bar{\delta} \\ 0 \leq \delta_t \leq 1, t=1, \dots, T}} \max_{\substack{\lambda: B^{tr} \lambda \leq \beta \\ \lambda \in \mathbb{R}_+^T}} \sum_{t=1}^T [(\bar{d}_t + \delta_t \Delta_t - (Ax)_t) \lambda_t] \end{array} \right.$$

où pour un vecteur (u) , nous notons par $(u)_t$, la t -ème composante de (u) . $DR(x)$ peut se réécrire :

$$DR(x) \left| \begin{array}{l} \max_{\lambda, \delta} \sum_{t=1}^T [(\bar{d}_t - (Ax)_t) \lambda_t + \Delta_t \delta_t \lambda_t] \\ B^{tr} \lambda \leq \beta \\ \sum_{t=1}^T \delta_t \leq \bar{\delta} \\ 0 \leq \delta_t \leq 1, t = 1, \dots, T \\ \lambda \in \mathbb{R}_+^T. \end{array} \right. \quad \begin{array}{l} (3.5) \\ (3.7) \\ (3.8) \\ (3.6) \end{array}$$

Cependant, ce programme bilinéaire avec des contraintes linéaires, n'est pas concave et il n'existe donc pas d'algorithmes efficaces pour calculer une solution optimale de $DR(x)$. Nous montrons maintenant que nous pouvons trouver une solution optimale de $DR(x)$ en résolvant un programme linéaire mixte équivalent. Pour cela, nous montrons d'abord la proposition suivante :

Proposition 2. *Il existe une solution optimale λ^*, δ^* de $DR(x)$ telle que $\delta_t^* \in \{0, 1\}$, $1 \leq t \leq T$.*

Démonstration. C'est une conséquence directe du lemme 1. En effet les points extrêmes du polytope \mathfrak{D} , sont des vecteurs où $\delta \in \{0, 1\}^T$. \square

Grace à la proposition ci-dessus, nous pouvons supposer qu'il existe une solution optimale de $DR(x)$, telle que $\lambda_t \delta_t$ appartienne à $\{0, \lambda_t\}$. Afin de pouvoir linéariser $\lambda_t \delta_t$, nous montrons maintenant que l'on peut se restreindre au cas où λ_t est borné, pour tout t , par une constante Λ .

Proposition 3. *Il existe $\Lambda > 0$ qui peut être calculé à partir de B et β tel que les contraintes $\lambda_t \leq \Lambda$, $t = 1, \dots, T$, peuvent être ajoutées à $DR(x)$ sans perte de généralité.*

Démonstration. Soit x réalisable. Réécrivons $DR(x)$ avec des variables d'écart $\lambda'_t \geq 0$, $t = 1, \dots, T$. les contraintes (3.5) deviennent : $B^{tr} \lambda + \lambda' = \beta$. Soit $(\lambda^*, \lambda'^*, \delta^*)$ une solution optimale de $DR(x)$ ainsi réécrit où (λ^*, λ'^*) est une solution de base de $DR(x)$ quand δ est fixé à δ^* .

Il existe donc une matrice de base $E = (e_{ij})$ de $(B^{tr} I_T)$ et des vecteurs de base

$\lambda_E^*, \lambda_E'^*$ tels que : $(\lambda_E^* \lambda_E'^*)^{tr} = E^{-1}\beta$. Soit \hat{e} une borne supérieure de la valeur absolue des coefficients de E^{-1} sur toutes les matrices de base E de $(B^{tr} I_T)$. Soit $\hat{\beta} = \max_{i=1, \dots, q} \beta_i$, nous avons $\lambda_t^* \leq \hat{e}\hat{\beta}q$, $t = 1, \dots, T$. Ainsi, il existe une solution optimale (λ^*, δ^*) de $DR(x)$ telle que λ_t^* est bornée par $\Lambda = \hat{e}\hat{\beta}q$ pour tout $t = 1, \dots, T$. \square

Nous pouvons maintenant linéariser $DR(x)$ en remplaçant les produits $\lambda_t \delta_t$ par de nouvelles variables ν_t et en ajoutant les contraintes de linéarisation :

$$\begin{aligned} \nu_t &\leq \lambda_t \\ \nu_t &\leq \Lambda \delta_t \\ \nu_t &\geq \lambda_t - \Lambda(1 - \delta_t) \\ \nu_t &\geq 0. \end{aligned}$$

$DR(x)$ est donc équivalent au programme linéaire mixte suivant :

$$LDR(x) \left\{ \begin{array}{l} \max_{\lambda, \delta, \nu} \sum_{t=1}^T [(\bar{d}_t - (Ax)_t)\lambda_t + \Delta_t \nu_t] \\ B^{tr} \lambda \leq \beta \\ \sum_{t=1}^T \delta_t \leq \bar{\delta} \\ \nu_t \leq \lambda_t, \quad t = 1, \dots, T \\ \nu_t \leq \Lambda \delta_t, \quad t = 1, \dots, T \\ \lambda, \nu \in \mathbb{R}_+^T \\ \delta_t \in \{0, 1\}, \quad t = 1, \dots, T. \end{array} \right.$$

Remarquons que les contraintes de linéarisation, $\nu_t \geq \lambda_t - \Lambda(1 - \delta_t)$, $t = 1, \dots, T$, peuvent être omises, puisque les coefficients des ν_t dans la fonction objectif à maximiser sont positifs.

3.3 L'algorithme de génération de contraintes

Afin de résoudre le problème robuste PR , nous le réécrivons d'abord comme un programme linéaire équivalent comportant un nombre exponentiel de contraintes, puis nous utilisons un algorithme de génération de contraintes pour le résoudre. Dans la section précédente, nous avons prouvé que la résolution du problème de recours était équivalente à la résolution du programme linéaire mixte $LDR(x)$. On peut donc réécrire le problème robuste en :

$$PR \left\{ \begin{array}{l} \min_x \alpha \cdot x + v(LDR(x)) \\ Cx \geq b \\ x_i \in \mathbb{N}, i = 1, \dots, p_1, x_i \in \mathbb{R}_+, i = (p_1 + 1), \dots, p. \end{array} \right.$$

Soit \mathcal{P}_Q le polyèdre défini par les contraintes de $LDR(x)$ où l'on a remplacé $\delta_t \in \{0, 1\}$ par $0 \leq \delta_t \leq 1$, et soit $(\mathcal{P}_Q)_I = conv(\mathcal{P}_Q \cap \{\delta \in \mathbb{N}^m\})$, l'enveloppe

convexe des solutions admissibles de $LDR(x)$. Remarquons que cette enveloppe convexe ne dépend pas de x . $(\mathcal{P}_Q)_I$ est un polyèdre et nous avons :

$$LDR(x) \left| \begin{array}{l} \max_{\lambda, \delta, \nu} \sum_{t=1}^T [(\bar{d}_t - (Ax)_t)\lambda_t + \Delta_t \nu_t] \\ \begin{pmatrix} \lambda \\ \delta \\ \nu \end{pmatrix} \in (\mathcal{P}_Q)_I, \end{array} \right.$$

Soit $\mathcal{S} = \{(\lambda^s, \delta^s, \nu^s)_{1 \leq s \leq S}\}$, l'ensemble des points extrêmes de $(\mathcal{P}_Q)_I$. Pour tout x réalisable, il existe un indice $s \in \{1, \dots, S\}$ tel que $(\lambda^s, \delta^s, \nu^s)$ est une solution optimale de $LDR(x)$.

On peut donc reformuler le problème robuste en le programme linéaire suivant :

$$PR \left| \begin{array}{l} \min_{x, z} \alpha x + z \\ z \geq \sum_{t=1}^T [(\bar{d}_t - (Ax)_t)\lambda_t^s + \Delta_t \nu_t^s], \quad 1 \leq s \leq S \\ Cx \geq b \\ x_i \in \mathbb{N}, \quad i = 1, \dots, p_1, \quad x_i \in \mathbb{R}_+, \quad i = (p_1 + 1), \dots, p, \quad z \in \mathbb{R} \end{array} \right. \quad (3.9)$$

Cependant, le nombre de contraintes du programme ci-dessus peut être exponentiel et nous ne connaissons pas à priori l'ensemble \mathcal{S} . Pour pallier ces problèmes, nous utilisons un algorithme de génération de contraintes, semblable à celui qui est utilisé dans [35], [18], ou encore dans [6] pour résoudre PR . Initialement, on considère un sous-ensemble \mathcal{S}^0 de \mathcal{S} ; à l'étape k , nous considérons un sous ensemble \mathcal{S}^k de \mathcal{S} et nous résolvons un programme relâché, PR^k , de PR , appelé *problème maître*, qui consiste à résoudre PR avec l'ensemble de contraintes (3.9) correspondant à \mathcal{S}^k . La solution obtenue est notée (x^k, z^k) .

Nous résolvons alors $LDR(x^k)$, appelé *programme esclave*, pour vérifier si (x^k, z^k) est une solution optimale. Si ce n'est pas le cas, une nouvelle contrainte est ajoutée au problème maître, i.e. un point extrême est ajouté à \mathcal{S}^k (Voir l'algorithme 1).

Montrons que cet algorithme converge.

En effet, il est clair que la valeur L d'une solution de PR^k à l'itération k représente une borne inférieure de la valeur de $v(PR)$, puisqu'à chaque itération les contraintes de PR^k ne représentent qu'un sous ensemble des contraintes de PR . Par ailleurs, à chaque itération, la solution $(x^k, \lambda^k, \delta^k, \nu^k)$ est une solution réalisable de PR^k mais pas de PR^{k+1} , puisque par définition une nouvelle contrainte, réalisable pour P , est ajoutée si et seulement si $L < U$, c'est à dire si

$$z^k < \sum_{t=1}^T (\bar{d}_t - (Ax)_t)\lambda_t^k + \Delta_t \nu_t^k.$$

Algorithm 1 Algorithme de génération de contraintes

1: $(\lambda^0, \delta^0, \nu^0) = (0, 0, 0)$. Soit $L \leftarrow -\infty, U \leftarrow +\infty, k \leftarrow 1$.

2: Résoudre le problème maître :

$$(PR)^k \left\{ \begin{array}{l} \min_{x,z} \alpha \cdot x + z \\ z \geq \sum_{t=1}^T (\bar{d}_t - (Ax)_t) \lambda_t^s + \Delta_t \nu_t^s, \quad 0 \leq s \leq k-1 \\ Cx \geq b \\ x_i \in \mathbb{N}, \quad i = 1, \dots, p_1, \quad x_i \in \mathbb{R}_+, \quad i = (p_1 + 1), \dots, p \\ z \in \mathbb{R} \end{array} \right.$$

Soit (x^k, z^k) la solution obtenue.

$$L \leftarrow \alpha x^k + z^k.$$

3: Résoudre $LDR(x^k)$. Soit $(\lambda^k, \delta^k, \nu^k)$ la solution optimale.

$$U \leftarrow \min\{U, \alpha x^k + v(LDR(x^k))\}.$$

Si $U = L$, **alors** retourner (x^k, z^k) **sinon** aller en 4.

4: Ajouter la contrainte

$$z \geq \sum_{t=1}^T (\bar{d}_t - (Ax)_t) \lambda_t^k + \Delta_t \nu_t^k,$$

au problème maître PR^k , $k \leftarrow k + 1$ et aller en 2.

La borne supérieure U consiste à prendre la meilleure solution admissible parmi celles qui sont générées jusqu'à l'itération k .

Remarquons que la valeur de L est non décroissante à chaque itération et montrons qu'à chaque itération k , la contrainte

$$z \geq \sum_{t=1}^T (\bar{d}_t - (Ax)_t) \lambda_t^k + \Delta_t \nu_t^k,$$

qui correspond au point extrême $(\lambda^k, \delta^k, \nu^k)$ de \mathcal{S} n'a pas déjà été ajoutée.

En effet, supposons que la contrainte ci-dessus soit ajoutée deux fois, aux itérations $k - q$ et k pour $q > 0$. Alors, à l'étape k , (x^k, z^k) doit vérifier :

$$z^k \geq \sum_{t=1}^T (\bar{d}_t - (Ax)_t) \lambda_t^{k-q} + \Delta_t \nu_t^{k-q},$$

puisque la contrainte a été ajoutée lors de l'itération $k - q$.

Or $(x^k, \lambda^{k-q}, \delta^{k-q}, \nu^{k-q})$ est une solution réalisable de PR^k , puisque la contrainte a été ajoutée lors de l'itération k , et qu'ainsi

$$(\lambda^{k-q}, \delta^{k-q}, \nu^{k-q}) = (\lambda^k, \delta^k, \nu^k).$$

Ainsi,

$$\alpha \cdot x^k + v(LDR(x^k)) \geq L = \alpha \cdot x^k + z^k.$$

Donc

$$z^k = \sum_{t=1}^T (\bar{d}_t - (Ax)_t) \lambda_t^{k-q} + \Delta_t \nu_t^{k-q}.$$

Ainsi $L = U$ et $(x^k, \lambda^{k-q}, \delta^{k-q}, \nu^{k-q})$ est une solution optimale de PR . Le nombre de points extrêmes de \mathcal{S} étant fini, nous déduisons de ce qui précède que l'algorithme converge vers la solution optimale du problème.

3.4 Quelques généralisations

Dans cette section, nous commencerons par généraliser les résultats des sections précédentes à un problème robuste plus général. Nous montrerons que l'on peut facilement généraliser l'approche précédente à un problème qui possède aussi des coefficients incertains dans le membre gauche des contraintes et comment étendre la résolution du problème robuste à un ensemble d'incertitude \mathcal{D} plus général.

3.4.1 Incertitude dans le membre gauche des contraintes

Montrons que nous pouvons généraliser notre approche au cas où les coefficients des contraintes ($A = (A_{ti})_{1 \leq t \leq T, 1 \leq i \leq p}$), sont également incertains. Comme précédemment, nous supposons que chaque coefficient A_{ti} appartient à l'intervalle $[\bar{A}_{ti} - \Gamma_{ti}, \bar{A}_{ti} + \Gamma_{ti}]$, où \bar{A}_{ti} est donné et où Γ_{ti} est une borne donnée sur l'incertitude de A_{ti} .

De plus, de manière similaire à \mathcal{D} , nous supposons que la somme totale des variations de la i ème colonne de A , $A_i = (A_{ti}, t = 1, \dots, T)$, est bornée. Ainsi, l'ensemble d'incertitude \mathcal{A}_i de A_i est défini par :

$$\mathcal{A}_i = \{A_i : A_{ti} = \bar{A}_{ti} - \gamma_{ti}\Gamma_{ti}, 0 \leq \gamma_{ti} \leq 1, t = 1, \dots, T, \sum_{t=1}^T \gamma_{ti} \leq \bar{\gamma}_i\},$$

où $\bar{\gamma}_i$ est un entier donné.

Le problème robuste s'écrit alors :

$$PR' \left| \begin{array}{l} \min_x \alpha x + \max_{\substack{A_i \in \mathcal{A}_i, \\ \forall i=1, \dots, p \\ d \in \mathcal{D}}} \min_y \beta y \\ \\ By \geq d - (A_1, \dots, A_p)x \\ y \in \mathbb{R}_+^q \\ \\ Cx \geq b \\ x_i \in \mathbb{N}, i = 1, \dots, p_1, x_i \in \mathbb{R}_+, i = (p_1 + 1), \dots, p. \end{array} \right.$$

Et le problème de recours devient :

$$DR'(x) \left| \begin{array}{l} \max_{\lambda, \delta, \gamma} \sum_{t=1}^T [(\bar{d}_t - \sum_{i=1}^p \bar{A}_{ti}x_i)\lambda_t + \Delta_t \delta_t \lambda_t + \sum_{i=1}^p \Gamma_{ti}x_i \gamma_{ti} \lambda_t] \\ B^{tr} \lambda \leq \beta \\ \sum_{t=1}^T \delta_t \leq \bar{\delta} \\ 0 \leq \delta_t \leq 1, \quad t = 1, \dots, T \\ \lambda \in \mathbb{R}_+^T \\ \sum_{t=1}^T \gamma_{ti} \leq \bar{\gamma}_i, \quad i = 1, \dots, p \\ 0 \leq \gamma_{ti} \leq 1, \quad i = 1, \dots, p, t = 1, \dots, T. \end{array} \right.$$

Nous pouvons alors linéariser les termes quadratiques ($\delta_t \lambda_t$ et $\gamma_{ti} \lambda_t$), pour obtenir un programme linéaire mixte en nombres entiers et résoudre le problème robuste comme nous l'avons fait dans les sections précédentes.

Considérons maintenant le cas où la matrice B a des coefficients incertains. Comme précédemment, nous supposons que pour, $t = 1, \dots, T$ et $i = 1, \dots, q$, $B_{ti} \in [\bar{B}_{ti} - \Phi_{ti}, \bar{B}_{ti} + \Phi_{ti}]$ et que la i ème colonne de B appartient à l'ensemble $\{B_i : B_{ti} = \bar{b}_{ti} - \phi_{ti}\Phi_{ti}, 0 \leq \phi_{ti} \leq 1, t = 1, \dots, T, \sum_{t=1}^T \phi_{ti} \leq \bar{\phi}_i\}$. Le dual du programme de recours est similaire à $DR(x)$ défini à la section 3.2 : des variables ϕ et les contraintes $\sum_{t=1}^T \phi_{ti} \leq \bar{\phi}_i, i = 1, \dots, q$ et $0 \leq \phi_{ti} \leq 1, i = 1, \dots, q, t = 1, \dots, T$, sont ajoutées à $DR(x)$ et nous remplaçons les contraintes 3.5 par : $\sum_{t=1}^T (\bar{B}_{ti} - \phi_{ti}\Phi_{ti})\lambda_t \leq \beta_i, i = 1, \dots, q$ 3.5'.

Il existe maintenant des termes quadratiques $(\phi_{ti}\lambda_t)$ dans les contraintes. Pour linéariser ces termes, nous devons vérifier que l'on peut étendre les propositions 2 et 3, c'est à dire, montrer qu'il existe une solution optimale telle que $\phi \in \{0, 1\}^{Tq}$ et que l'on peut calculer à priori une borne de λ . En effet, les preuves données à la section 3.2 peuvent s'étendre facilement. Soit λ fixé. Si les contraintes 3.5' et $\sum_{t=1}^T \phi_{ti} \leq \bar{\phi}_i, i = 1, \dots, q$ sont vérifiées, une solution entière réalisable peut être obtenue en fixant ϕ_{ti} à 1, si i correspond à l'une des $\bar{\phi}_i$ plus grandes valeurs de $\Phi_{ti}\lambda_t$, et à 0 sinon.

Quand à la proposition 3, en utilisant les mêmes arguments qu'à la section 3.2, pour tout ϕ fixé, nous pouvons borner λ_t par $\Lambda(\phi)$, qui, similairement à Λ , borne tout point extrême du polyèdre des contraintes introduit dans la preuve de la proposition 3, lorsque ϕ est fixé. Comme $\phi_{ti} \in \{0, 1\}$ et $\sum_{t=1}^T \phi_{ti} \leq \bar{\phi}_i$, il existe 2^{qT} valeurs différentes de ϕ et nous pouvons borner λ_t par $\max_{\phi} \Lambda(\phi)$.

Finalement lorsque tous les paramètres sont incertains, le programme de recours linéarisé se réécrit :

$$\begin{array}{l}
\max_{\lambda, \delta, \gamma, \phi} \quad \sum_{t=1}^T [(d_t - \sum_{i=1}^p \bar{A}_{ti}x_i)\lambda_t + \Delta_t\nu_t + \sum_{i=1}^p \Gamma_{ti}x_i\mu_{ti}] \\
\left. \begin{array}{l}
\sum_{t=1}^T \bar{B}_{ti}\lambda_t - \sum_{t=1}^T \Phi_{ti}\mu_{ti} \leq \beta_i, \quad i = 1, \dots, q \\
\sum_{t=1}^T \delta_t \leq \bar{\delta} \\
0 \leq \delta_t \leq 1, \quad t = 1, \dots, T \\
\sum_{t=1}^T \gamma_{ti} \leq \bar{\gamma}_i, \quad i = 1, \dots, p \\
0 \leq \gamma_{ti} \leq 1, \quad i = 1, \dots, p, t = 1, \dots, T \\
\sum_{t=1}^T \phi_{ti} \leq \bar{\phi}_i, \quad i = 1, \dots, q \\
0 \leq \phi_{ti} \leq 1, \quad i = 1, \dots, q, t = 1, \dots, T \\
\nu_t \leq \lambda_t, \quad t = 1, \dots, T \\
\nu_t \leq \Lambda\delta_t, \quad t = 1, \dots, T \\
\delta_t \in \{0, 1\}, \quad t = 1, \dots, T \\
\mu_{ti} \leq \lambda_t, \quad t = 1, \dots, T, i = 1, \dots, p \\
\mu_{ti} \leq \Lambda\gamma_{ti}, \quad t = 1, \dots, T, i = 1, \dots, p \\
\gamma_{ti} \in \{0, 1\}, \quad t = 1, \dots, T, i = 1, \dots, p \\
\rho_{ti} \leq \lambda_t, \quad t = 1, \dots, T, i = 1, \dots, q \\
\rho_{ti} \leq \Lambda\phi_{ti}, \quad t = 1, \dots, T, i = 1, \dots, q \\
\rho_{ti} \in \{0, 1\}, \quad t = 1, \dots, T, i = 1, \dots, q \\
\lambda, \nu \in \mathbb{R}_+^T \\
\mu_{ti}, \rho_{tj} \in \mathbb{R}_+, \quad t = 1, \dots, T, i = 1, \dots, p, j = 1, \dots, q.
\end{array} \right\} LDR'(x)
\end{array}$$

où $\mu_{ti} = \lambda_t \gamma_{ti}$, et $\rho_{ti} \leq \lambda_t \phi_{ti}$, pour tout $t = 1, \dots, T$, $i = 1, \dots, q$.

3.4.2 Généralisation à d'autres ensembles d'incertitudes

Dans les parties précédentes, nous avons supposé que les coefficients incertains s'écrivaient $d_t = \bar{d}_t + \delta_t \Delta_t \forall t$, où les variables δ_t expriment l'incertitude et vérifient $\sum_{t=1}^T \delta_t \leq \bar{\delta}$. Nous montrons maintenant que nous pouvons généraliser nos résultats au cas où d peut s'écrire $d = \bar{d} + D\delta$, où le vecteur \bar{d} et la matrice D sont donnés et où δ appartient à un polytope \mathcal{D} dont les points extrêmes (d^1, \dots, d^S) sont connus. Remarquons que cette définition de l'incertitude englobe celle de Babonneau et al. dans [7]. Réécrivons le problème de recours :

$$DR'(x) \left| \begin{array}{l} \max_{\lambda, \delta} (\bar{d} + D\delta - Ax) \cdot \lambda \\ B^{tr} \lambda \leq \beta \\ \delta \in \mathcal{D} \\ \lambda \in \mathbb{R}_+^T. \end{array} \right.$$

Soit $v^1, \dots, v^S \in [0, 1]$ les variables telles que $\delta = \sum_{s=1}^S d^s v^s$ et $\sum_{s=1}^S v^s = 1$. Nous pouvons alors réécrire le problème de recours :

$$DR'(x) \left| \begin{array}{l} \max_{\lambda, v} (\bar{d} - Ax) \cdot \lambda + \sum_{s=1}^S (v^s (Dd^s) \cdot \lambda) \\ B^{tr} \lambda \leq \beta \\ \sum_{s=1}^S v^s = 1 \\ 0 \leq v^s \leq 1, \quad s = 1, \dots, S \\ \lambda \in \mathbb{R}_+^T. \end{array} \right.$$

En utilisant les mêmes arguments que dans la preuve de la proposition 2, on montre qu'il existe une solution optimale (λ^*, v^*) de $DR'(x)$ telle que $v^{s^*} = 1$ ou $v^{s^*} = 0$ pour tout $s = 1, \dots, S$ $s \neq s^*$. Nous pouvons alors linéariser les termes $v^s \lambda_t$, pour tout s et pour tout t , comme nous l'avons fait en 3.2 et obtenir un problème linéaire mixte.

Chapitre 4

Résolution du problème robuste sans la propriété de recours complet

Dans le chapitre précédent, nous avons supposé que le problème déterministe satisfaisait la propriété \mathcal{P} . Nous montrons maintenant que nous pouvons étendre nos résultats au cas où nous supposons seulement que le problème robuste PR admet une solution, c'est-à-dire qu'il existe x tel que pour tout d dans \mathcal{D} , il existe une solution réalisable y de $R(x, d)$. Cependant nous montrerons que notre méthode permet aussi de détecter les cas où le problème robuste n'a pas de solution. Par souci de clarté, avant de donner la preuve dans le cas général, nous traitons le cas où toutes les variables de décisions sont entières. Dans les deux cas, nous considérons un nouveau programme déterministe, P_ε , vérifiant la propriété \mathcal{P} , où de nouvelles variables de recours w_t affectées d'un coefficient dépendant de ε dans la fonction objectif, ont été ajoutées. Nous montrons alors que lorsque ε est suffisamment petit, toute solution optimale du programme robuste associé à P_ε est aussi une solution optimale de PR . Nous testons ensuite notre approche sur un problème de production dans lequel une entreprise décide de construire des usines pour fabriquer plusieurs produits afin de répondre à une demande sur chacun de ces produits.

4.1 Résolution du problème

Dans cette section, nous construirons un nouveau programme linéaire mixte P_ε , qui vérifie la propriété \mathcal{P} . Nous montrerons en outre que si PR a une solution optimale, alors toute solution optimale du nouveau problème robuste associé à P_ε est aussi une solution optimale de PR . Pour obtenir P_ε , nous ajoutons de nouvelles variables positives non bornées w_t , $t = 1, \dots, T$, au problème initial P . Comme dans les sections 3.1 et 3.2, par souci de clarté et sans perte de généralité, nous supposons que les coefficients incertains sont uniquement dans le membre droit des contraintes.

4.1.1 Modélisation

Supposons que le problème robuste PR admette une solution. Il existe alors un vecteur x réalisable pour tout $d \in \mathcal{D}$. Le problème déterministe P admet donc une solution optimale quel que soit le second membre $d \in \mathcal{D}$.

Puisque le problème robuste PR admet une solution optimale, il existe un réel U tel que $v(PR) \leq U$. Soit L tel que pour tout $d \in \mathcal{D}$, $v(P) \geq L$.

Avant de présenter le nouveau problème robuste que nous étudierons, nous montrons comment calculer L et U .

Remarquons que puisque toutes les données, $A, B, C, b, d, \alpha, \beta, \Delta$, de PR ont des coefficients rationnels, nous pouvons nous ramener au cas où tous les coefficients sont des nombres entiers. Nous supposons donc dans le reste de ce chapitre que tous les programmes mathématiques ont pour coefficients des nombres entiers.

On note $\|\cdot\|_\infty$ la norme infinie d'un vecteur v ou d'une matrice A . C'est à dire, le plus grand élément, en valeur absolue, du vecteur ou de la matrice.

Proposition 4. *La valeur d'une solution optimale de P est minorée par*

$$L = -(p+q)(T+n)(T+n-1)^{\frac{T+n-1}{2}} \|(\alpha, \beta)\|_\infty \max(\|A\|_\infty, \|B\|_\infty, \|C\|_\infty)^{T+n-1} \|(d, b)\|_\infty,$$

Démonstration. Soit \bar{P} le programme linéaire relâché de P où toutes les variables sont continues. Une solution optimale de P étant bornée inférieurement, nous savons, d'après le théorème 5.2 de [26, chap. 5.1], qu'une solution optimale de \bar{P} est aussi bornée inférieurement.

Réécrivons les contraintes de \bar{P} avec des variables d'écart positives e et f . Soit

$$H = \begin{pmatrix} A & B & -I_T & 0 \\ C & 0 & 0 & -I_n \end{pmatrix}, \text{ les contraintes de } \bar{P} \text{ se réécrivent :}$$

$$H \begin{pmatrix} x \\ y \\ e \\ f \end{pmatrix} = \begin{pmatrix} d \\ b \end{pmatrix}$$

Soit (x^*, y^*, e^*, f^*) une solution optimale de base de \bar{P} . Il existe une matrice de base $H_B \in \mathbb{Z}^{(T+n) \times (T+n)}$ extraite de H et un vecteur $(x_B^*, y_B^*, e_B^*, f_B^*)$ tels que

$$(x_B^*, y_B^*, e_B^*, f_B^*)^{tr} = H_B^{-1} \begin{pmatrix} d \\ b \end{pmatrix} = \frac{1}{\det(H_B)} \text{com}(H_B)^{tr} \begin{pmatrix} d \\ b \end{pmatrix}$$

où $\text{com}(H_B)$ est la comatrice de H_B .

Remarquons que si Z est une matrice à n colonnes, alors $\|Zz\|_\infty \leq n\|Z\|_\infty\|z\|_\infty$,

pour tout vecteur z . Ainsi $\|(x, y)\|_\infty \leq \frac{1}{|\det(H_B)|} (T+n) \|com(H_B)^{tr}\|_\infty \|(d, b)\|_\infty$. Tous les coefficients de H_B étant des entiers, $|\det(H_B)| \geq 1$, en outre tous les coefficients de $com(H_B)$ sont des déterminants de sous matrices, de taille $(T+n-1) \times (T+n-1)$, extraites de H_B . D'après l'inégalité d'Hadamard, on a $\|com(H_B)^{tr}\|_\infty \leq \|H\|_\infty^{T+n-1} (T+n-1)^{\frac{T+n-1}{2}}$. Nous avons donc

$$v(P) \geq v(\bar{P}) \geq \alpha.x^* + \beta.y^* \geq -(p+q) \|(\alpha, \beta)\|_\infty \|x^*, y^*\|_\infty.$$

d'où

$$v(P) \geq -(p+q) \|(\alpha, \beta)\|_\infty (T+n) \|H\|_\infty^{T+n-1} (T+n-1)^{\frac{T+n-1}{2}} \|(d, b)\|_\infty.$$

Tous les coefficients de A, B et C étant des entiers, nous avons $\|H\|_\infty = \max(\|A\|_\infty, \|B\|_\infty, \|C\|_\infty)$, ce qui achève la preuve. \square

Nous donnons maintenant une valeur générale pour U .

D'après le lemme 1 le scénario extrême associé à un vecteur x réalisable est un point extrême de \mathcal{D} . Soit $\{d^1, \dots, d^S\}$, l'ensemble des points extrêmes de \mathcal{D} . Remarquons que puisque $d_t^s = \bar{d}_t$ où $d_t^s = \bar{d}_t + \Delta_t$, pour tout s et pour tout t , d^s est un vecteur entier pour tout s . Nous pouvons réécrire le problème robuste PR en le programme linéaire mixte suivant où y^s est la variable de recours associée au scénario s :

$$PR \left\{ \begin{array}{l} \min_{x,z} \alpha.x + z \\ y^1, \dots, y^S \\ z \geq \beta.y^s, \quad s = 1, \dots, S \\ Ax + By^s \geq d^s, \quad s = 1, \dots, S \\ Cx \geq b \\ x_i \in \mathbb{N}, \quad i = 1, \dots, p_1, \quad x_j \in \mathbb{R}_+, \quad j = p_1 + 1, \dots, p \\ y^s \in \mathbb{R}_+^q, \quad s = 1, \dots, S \end{array} \right. \quad (4.1)$$

$$(4.2)$$

En ajoutant les variables d'écart, $e^s, f^s, s = 1, \dots, S$ et g , nous pouvons réécrire les contraintes du programme ci-dessus en $Hu = v$ où

$$H = \begin{pmatrix} 0 & 1 & -\beta & -1 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -\beta & -1 & 0 & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots & \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & -\beta & -1 & 0 & 0 \\ A & 0 & B & 0 & -I_T & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ A & 0 & 0 & 0 & 0 & B & 0 & -I_T & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots & \\ A & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & B & 0 & -I_T & 0 \\ C & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & -I_n \end{pmatrix},$$

$$u^{tr} = (x, z, y^1, e^1, f^1, \dots, y^S, e^S, f^S, g)$$

et

$$v^{tr} = (0, \dots, 0, d^1, \dots, d^S, b).$$

Soit le problème relâché $\bar{P}R$ de PR , dont les variables, notées \bar{u} , sont dans \mathbb{R}_+ . Soit \bar{u}^* un point extrême de $\bar{P}R$. Il existe alors une $(S + ST + n) \times (S + ST + n)$ -matrice de base H_B de H et un vecteur de base \bar{u}_B^* tels que

$$\bar{u}_B^* = \frac{1}{\det(H_B)} \text{com}^{tr}(H_B)v.$$

Puisque tous les coefficients de H sont des entiers, $|\det(H_B)| \geq 1$, ainsi

$$\|\bar{u}_B^*\|_\infty \leq (S + ST + n) \|\text{com}^{tr}(H_B)\|_\infty \|v\|_\infty.$$

De plus, comme tous les coefficients de $\text{com}^{tr}(H_B)$ sont des déterminants de matrices de taille $(S + ST + n - 1) \times (S + ST + n - 1)$, nous savons, d'après l'inégalité d'Hadamard que :

$$\|\text{com}^{tr}(H_B)\|_\infty \leq \|H\|_\infty^{S+ST+n-1} (S + ST + n - 1)^{\frac{S+ST+n-1}{2}}.$$

Sachant que $S \leq \left(\frac{T}{\delta}\right) + 1$, $\|H\|_\infty = \max(\|A\|_\infty, \|B\|_\infty, \|C\|_\infty, \|\beta\|_\infty)$ et que $\|v\|_\infty = \max(\|\bar{d}\|_\infty + \|\Delta\|_\infty, \|b\|_\infty)$, nous avons

$$\alpha \bar{x}^* + \bar{z}^* \leq p \|\alpha\|_\infty \|\bar{u}^*\|_\infty + \|\bar{u}^*\|_\infty,$$

et ainsi

$$\alpha \bar{x}^* + \bar{z}^* \leq (p \|\alpha\|_\infty + 1) (S + ST + n) \|H\|_\infty^{S+ST+n-1} (S + ST + n - 1)^{\frac{S+ST+n-1}{2}} \|v\|_\infty.$$

Or, d'après le théorème 5.6 de [26, chap. 5.1], il existe une solution réalisable u du problème robuste telle que $\|u - \bar{u}^*\|_\infty \leq (S + ST + n) \Xi(H)$, où $\Xi(H)$ est le maximum des sous-déterminants de H . Ainsi $\Xi(H)$ peut être majoré en utilisant à nouveau l'inégalité d'Hadamard.

Ainsi $v(PR) \leq U = (p \|\alpha\|_\infty + 1) \|u\|_\infty \leq (p \|\alpha\|_\infty + 1) (\|\bar{u}^*\|_\infty + \|u - \bar{u}^*\|_\infty)$.

Remarquons que les deux bornes L et U , calculées ci-dessus, ne sont pas très faciles à manier. Cependant, suivant le problème rencontré, de meilleures valeurs peuvent être calculées, comme c'est le cas dans l'application présentée en 4.2.

Nous pouvons maintenant définir un nouveau problème déterministe qui satisfait la propriété \mathcal{P} . Soit ε un réel strictement positif donné, nous définissons le programme linéaire mixte suivant :

$$\begin{array}{l}
P_\varepsilon \left\{ \begin{array}{l}
\min_{x,y,w} \alpha x + \beta y + \frac{U-L}{\varepsilon} \sum_{t=1}^T w_t \\
Ax + By + w \geq d \quad (1\varepsilon) \\
\alpha \cdot x + \beta \cdot y \geq L \quad (5\varepsilon) \\
Cx \geq b \quad (2) \\
x_i \in \mathbb{N}, i = 1, \dots, p_1, x_i \in \mathbb{R}_+, i = (p_1 + 1), \dots, p \quad (3) \\
y \in \mathbb{R}_+^q, w \in \mathbb{R}_+^T \quad (4\varepsilon)
\end{array} \right.
\end{array}$$

Comme les variables w_t , $t = 1, \dots, T$, ne sont pas bornées, P_ε vérifie la propriété \mathcal{P} . Remarquons en outre qu'à partir de toute solution admissible (x, y) de P , on peut construire une solution $(x, y, 0)$ admissible de P_ε , la contrainte 5ε étant vérifiée par n'importe quelle solution admissible de P quel que soit $d \in \mathcal{D}$.

Soit PR_ε , le problème robuste associé à P_ε ,

$$\begin{array}{l}
PR_\varepsilon \left\{ \begin{array}{l}
\min_x \alpha x + \max_{d \in \mathcal{D}} \min_{y,w} \beta y + \frac{M}{\varepsilon} \sum_{t=1}^T w_t \\
By + w \geq d - Ax \\
\beta \cdot y \geq L - \alpha \cdot x \\
y \in \mathbb{R}_+^q, w \in \mathbb{R}_+^T \\
Cx \geq b \\
x_i \in \mathbb{N}, i = 1, \dots, p_1, x_i \in \mathbb{R}_+, i = (p_1 + 1), \dots, p
\end{array} \right.
\end{array}$$

où $M = U - L$. Nous noterons en outre $R_\varepsilon(x)$, $R_\varepsilon(x, d)$ et $DR_\varepsilon(x)$ les sous-problèmes associés comme ceux définis dans la section 3.2.

4.1.2 Résolution

Proposition 5. $v(PR_\varepsilon)$ vérifie $v(PR_\varepsilon) \leq v(PR) \leq U$.

Démonstration. Soit (\hat{x}, \hat{y}) une solution optimale de PR , Par hypothèse, $v(PR) \leq U$ donc $v(R(\hat{x})) \leq U - \alpha \cdot \hat{x}$. Soit \bar{d} un scénario de \mathcal{D} . Puisque $v(R(\hat{x})) = \max_{d \in \mathcal{D}} v(R(\hat{x}, d))$, nous avons $v(R(\hat{x}, \bar{d})) \leq U - \alpha \cdot \hat{x}$. Soit \bar{y} une solution optimale de $R(\hat{x}, \bar{d})$, alors $(y, w) = (\bar{y}, 0)$ est une solution réalisable de même coût de $(R_\varepsilon(\hat{x}, \bar{d}))$. Ainsi $v(R_\varepsilon(\hat{x}, \bar{d})) \leq v(R(\hat{x}, \bar{d}))$, pour tout $\bar{d} \in \mathcal{D}$, ce qui implique $v(R_\varepsilon(\hat{x})) \leq v(R(\hat{x}))$, et $v(PR_\varepsilon) \leq v(PR) \leq U$. \square

Soit (x^*, d^*, y^*, w^*) une solution optimale de PR_ε où d^* est le scénario extrême associé à x^* . Remarquons que la proposition 2 est valide pour $DR_\varepsilon(x^*)$. Ainsi $d_t^* = \bar{d}_t$ ou $d_t^* = \bar{d}_t + \Delta_t$, et d_t^* est donc un entier. D'après la proposition 5 nous avons

$$\alpha x^* + \beta y^* + \frac{U-L}{\varepsilon} \sum_{t=1}^T w_t^* \leq U.$$

Or puisque $\alpha x^* + \beta y^* \geq L$, nous avons $\sum_{t=1}^T w_t^* \leq \varepsilon$, ainsi $w_t^* \leq \varepsilon, \forall t = 1, \dots, T$. Le reste de cette section consiste à montrer que, si le problème robuste initial admet une solution optimale, alors, lorsque ε est inférieur à une certaine valeur, $w^* = 0$ dans toute solution optimale, (y^*, w^*) , de $(R_\varepsilon(x^*, d^*))$, et qu'ainsi d'après la proposition 5 $v(PR) = v(PR_\varepsilon)$. Par souci de clarté, nous nous intéresserons, avant de traiter le cas général, au cas où toutes les variables de décision $x_i, i = 1, \dots, p$, sont entières.

Supposons que les variables x du problème robuste soient entières. Réécrivons le problème PR_ε avec les variables d'écart $\sigma = (\sigma_t, t = 1, \dots, T)$ et γ : les contraintes (1ε) deviennent $By + w - \sigma = d - Ax$, et la contrainte (5ε) devient $\beta \cdot y - \gamma = L - \alpha \cdot x$. Soit $(x^*, d^*, y^*, w^*, \sigma^*, \gamma^*)$ une solution optimale de PR_ε ainsi réécrit. Alors $(y^*, w^*, \sigma^*, \gamma^*)$ soit une solution optimale du programme

$$R_\varepsilon(x^*, d^*) \left| \begin{array}{l} \min_{y, w, \sigma, \gamma} \beta y + \frac{M}{\varepsilon} \sum_{t=1}^T w_t \\ By + w - \sigma = d^* - Ax^* \\ \beta \cdot y - \gamma = L - \alpha \cdot x^* \\ y \in \mathbb{R}_+^q, \sigma, w \in \mathbb{R}_+^T. \end{array} \right.$$

qui peut se réécrire

$$R_\varepsilon(x^*, d^*) \left| \begin{array}{l} \min_{y, w, \sigma, \gamma} \beta y + \frac{M}{\varepsilon} \sum_{t=1}^T w_t \\ \begin{pmatrix} B & I_T & -I_T & 0 \\ \beta & 0 & 0 & -1 \end{pmatrix} \begin{pmatrix} y \\ w \\ \sigma \\ \gamma \end{pmatrix} = d^* - A'x^* \\ y \in \mathbb{R}_+^q, \sigma, w \in \mathbb{R}_+^T, \gamma \in \mathbb{R} \end{array} \right.$$

où $d' = \begin{pmatrix} d \\ L \end{pmatrix}$, et $A' = \begin{pmatrix} A \\ \alpha \end{pmatrix}$.

Soit $H = \begin{pmatrix} B & I_T & -I_T & 0 \\ \beta & 0 & 0 & -1 \end{pmatrix} \in \mathbb{Z}^{(T+1) \times (q+2T+1)}$. Remarquons que H est de rang $T + 1$.

Proposition 6. Soit $\varepsilon < \frac{1}{\|H\|_\infty^{T+1} (T+1)^{(T+1)/2}}$. Si (x^*, d^*, y^*, w^*) est une solution optimale de PR_ε alors $w^* = 0$ et (x^*, d^*, y^*) est une solution optimale de PR .

Démonstration. Supposons d'abord que $(y^*, w^*, \sigma^*, \gamma^*)$ est une solution optimale de base de $R_\varepsilon(x^*, d^*)$.

Il existe une matrice de base $H_B = (h_{kj}) \in \mathbb{Z}^{(T+1) \times (T+1)}$ de H et un vecteur de base $(y_B^*, w_B^*, \sigma_B^*, \gamma_B^*)$ tels que : $H_B (y_B^* \ w_B^* \ \sigma_B^* \ \gamma_B^*)^{tr} = d^* - A'x^*$.

La matrice H_B est inversible, et $H_B^{-1} = \frac{1}{\det(H_B)} \text{com}^{tr}(H_B)$. Ainsi $(y_B^* \ w_B^* \ \sigma_B^* \ \gamma_B^*)^{tr} = \frac{1}{\det(H_B)} \text{com}^{tr}(H_B)(d'^* - A'x^*)$, et

$$0 \leq w_t^* = \frac{1}{\det(H_B)} (\text{com}^{tr}(H_B)(d'^* - A'x^*))_{t'} \leq \varepsilon,$$

où w_t^* est une variable de base où t' est l'indice associé dans $(y_B^*, w_B^*, \sigma_B^*, \gamma_B^*)$. Ainsi

$$|\text{com}^{tr}(H_B)(d'^* - A'x^*)_{t'}| \leq \varepsilon |\det(H_B)|. \quad (4.3)$$

D'après l'inégalité d'Hadamard : $|\det(H_B)| \leq \prod_{j=1}^{T+1} \sqrt{\sum_{k=1}^{T+1} h_{kj}^2}$. Et donc $|\det(H_B)| \leq \|H\|_{\infty}^{T+1} (T+1)^{(T+1)/2}$. Supposons que $\varepsilon < \frac{1}{\|H\|_{\infty}^{T+1} (T+1)^{(T+1)/2}}$, alors d'après (4.3), $|(\text{com}^{tr}(H_B)(d'^* - A'x^*))_{t'}| < 1$. Or, comme $|(\text{com}^{tr}(H_B)(d'^* - A'x^*))_{t'}| \in \mathbb{N}$, nous avons $|(\text{com}^{tr}(H_B)(d'^* - A'x^*))_{t'}| = 0$. Ainsi $w_t = 0$ pour toute variable de base w_t et donc $w_t = 0$ quel que soit $t = 1, \dots, T$.

Ainsi, en prenant $\varepsilon < \frac{1}{\|H\|_{\infty}^{T+1} (T+1)^{(T+1)/2}}$, toute solution de base optimale $(y^*, w^*, \sigma^*, \gamma^*)$ de $R_{\varepsilon}(x^*, d^*)$ doit vérifier $w^* = 0$.

Supposons maintenant que $(y^*, w^*, \sigma^*, \gamma^*)$ soit une solution optimale quelconque de $R_{\varepsilon}(x^*, d^*)$. Nous pouvons écrire $(y^*, w^*, \sigma^*, \gamma^*)$ comme la somme d'une combinaison convexe de solutions optimales de bases (vérifiant $w^{s*} = 0$ pour tout s) et d'une combinaison positive de rayons extrêmes du polyèdre des contraintes de $R_{\varepsilon}(x^*, d^*)$. Or, puisque toute solution optimale de $R_{\varepsilon}(x^*, d^*)$ doit vérifier $w^{s*} \leq \varepsilon$ pour tout $s = 1, \dots, S$, aucun rayon extrême dans la décomposition ne peut avoir $w_t^{s*} \neq 0$ pour un couple (s, t) avec $s \in \{1, \dots, S\}$ et $t \in \{1, \dots, T\}$.

Finalement, nous avons bien montré que pour toute solution $(x^*, y^*, w^*, \sigma^*, \gamma^*)$ de PR_{ε} , nous avons $w^* = 0$, et qu'ainsi (x^*, d^*, y^*) est une solution admissible de PR . Ainsi d'après la proposition 5, $v(PR) = v(PR_{\varepsilon})$ et (x^*, d^*, y^*) est une solution optimale de PR . \square

Nous revenons maintenant au cas général où les variables de décisions sont mixtes : $x_i \in \mathbb{N}$, $i = 1, \dots, p_1$, $x_i \in \mathbb{R}_+$, $i = (p_1 + 1), \dots, p$. D'après la proposition 2 le scénario extrême associé à un vecteur x réalisable est un point extrême d^s de \mathcal{D} . Comme dans le calcul de la borne U , nous pouvons réécrire PR_{ε} en le programme linéaire mixte suivant :

$$\begin{array}{l}
\min_{x,z} \quad \alpha_1 x^1 + \alpha_2 x^2 + z \\
\begin{array}{l} y^1, \dots, y^S \\ w^1, \dots, w^S \end{array} \\
\left. \begin{array}{l} z \geq \beta y^s + \frac{M}{\varepsilon} \sum_{t=1}^T w_t^s, \quad s = 1, \dots, S \\ A_1 x^1 + A_2 x^2 + B y^s + w^s \geq d^s, \quad s = 1, \dots, S \\ \alpha_1 \cdot x^1 + \alpha_2 \cdot x^2 + \beta \cdot y^s \geq L, \quad s = 1, \dots, S \\ C_1 x^1 + C_2 x^2 \geq b \\ x^1 \in \mathbb{N}^{p_1}, \quad x^2 \in \mathbb{R}_+^{p-p_1} \\ y^s \in \mathbb{R}_+^q, \quad w^s \in \mathbb{R}_+^T, \quad s = 1, \dots, S \end{array} \right\} PR'_\varepsilon
\end{array} \quad (4.4)$$

où $x = \begin{pmatrix} x^1 \\ x^2 \end{pmatrix}$, $\alpha = (\alpha_1, \alpha_2)$, $A = (A_1, A_2)$ et $C = (C_1, C_2)$.

Soit $(x^*, z^*, y^{1*}, \dots, y^{S*}, w^{1*}, \dots, w^{S*})$ une solution optimale de ce programme ; il existe $\hat{s} \in \{1, \dots, S\}$ tel que la contrainte 4.4 soit saturée :
 $z^* = \beta y^{\hat{s}*} + \frac{M}{\varepsilon} \sum_{t=1}^T w_t^{\hat{s}*}$ et \hat{s} est le pire scénario associé à x^* .

Le schéma de la preuve est le suivant : nous montrons d'abord que lorsque ε est suffisamment petit, nous avons $w^s = 0$ pour tout $s = 1, \dots, S$ dans toute solution optimale de PR'_ε , nous en déduisons alors que $w = 0$ dans toute solution optimale de PR_ε . Et finalement de toute solution optimale PR_ε nous obtenons une solution optimale de PR . Dans toute la suite nous poserons $l_M = \max(\|\alpha\|_\infty, \|\beta\|_\infty, \|A\|_\infty, \|B\|_\infty, \|C\|_\infty, L)$

Proposition 7. Si $\varepsilon < \frac{1}{l_M^{(ST+S+n)}(ST+S+n)^{(ST+S+n)/2}}$ alors toute solution de PR'_ε vérifie $w^s = 0$, pour tout $s = 1, \dots, S$.

Démonstration. Soit $(x^{1*}, x^{2*}, z^*, y^{1*}, \dots, y^{S*}, w^{1*}, \dots, w^{S*})$ une solution optimale de PR'_ε . d'après les contraintes 4.4 et la proposition 5, qui est aussi vraie pour des variables mixtes, nous avons pour tout $s = 1, \dots, S$:

$$\alpha_1 x^{1*} + \alpha_2 x^{2*} + \beta y^{s*} + \frac{U-L}{\varepsilon} \sum_{t=1}^T w_t^{s*} \leq \alpha_1 x^{1*} + \alpha_2 x^{2*} + z^* \leq U.$$

de plus d'après la contrainte 4.6, nous savons que $\frac{U-L}{\varepsilon} \sum_{t=1}^T w_t^{s*} \leq U-L$, donc

$$w_t^{s*} \leq \varepsilon, \quad \forall t, s. \quad (4.7)$$

Remarquons que $(x^{2*}, z^*, y^{1*}, \dots, y^{S*}, w^{1*}, \dots, w^{S*})$ est une solution optimale de $PR'_\varepsilon(x^{1*})$, qui correspond à PR'_ε lorsque x_1 est fixé à x_1^* . En dualisant la

contrainte 4.4, dans $PR'_\varepsilon(x^{1*})$, nous obtenons :

$$D'_\varepsilon(x^{1*}) \left| \begin{array}{l} \max_{\lambda \geq 0} \min_{\substack{x^2, z \\ y^1, \dots, y^S \\ w^1, \dots, w^S}} \alpha_2 x^2 + z + \sum_{s=1}^S \lambda_s \left(\beta y^s + \frac{M}{\varepsilon} \sum_{t=1}^T w_t^s - z \right) \\ A_2 x^2 + B y^s + w^s \geq d^s - A_1 x^{1*}, \quad s = 1, \dots, S \\ \alpha_2 \cdot x^2 + \beta \cdot y^s \geq L - \alpha_1 \cdot x^{1*}, \quad s = 1, \dots, S \\ C_2 x^2 \geq b - C_1 x^{1*} \\ x^2 \in \mathbb{R}_+^{p-p_1} \\ y^s \in \mathbb{R}_+^q, \quad w^s \in \mathbb{R}_+^T, \quad s = 1, \dots, S \end{array} \right.$$

qui vérifie $v(D'_\varepsilon(x^{1*})) = v(PR'_\varepsilon(x^{1*})) = v(PR'_\varepsilon)$.

Soit $(\lambda', x^{2'}, z', y^{1'}, \dots, y^{S'}, w^{1'}, \dots, w^{S'})$ une solution optimale de $D'_\varepsilon(x^{1*})$, alors $(x^{2'}, z', y^{1'}, \dots, y^{S'}, w^{1'}, \dots, w^{S'})$ est une solution optimale de $D'_\varepsilon(x^{1*}, \lambda')$ qui correspond au sous-programme de minimisation de $D'_\varepsilon(x^{1*})$ obtenu en fixant λ à λ' .

En ajoutant les variables d'écarts $e^s, f^s, s = 1, \dots, S$ et g , nous pouvons ré-écrire $D'_\varepsilon(x^{1*}, \lambda')$ en

$$D'_\varepsilon(x^{1*}, \lambda') \left| \begin{array}{l} \min_{\substack{x^2, z \\ y^1, \dots, y^S \\ w^1, \dots, w^S \\ e^1, \dots, e^S \\ f^1, \dots, f^S, g}} \alpha_2 x^2 + z + \sum_{s=1}^S \lambda'_s \left(\beta y^s + \frac{M}{\varepsilon} \sum_{t=1}^T w_t^s - z \right) \\ A_2 x^2 + B y^s + w^s - e^s = d^s - A_1 x^{1*}, \quad s = 1, \dots, S \\ \alpha_2 \cdot x^2 + \beta \cdot y^s - f^s = L - \alpha_1 \cdot x^{1*}, \quad s = 1, \dots, S \\ C_2 x^2 - g = b - C_1 x^{1*} \\ x^2 \in \mathbb{R}_+^{p-p_1} \\ y^s \in \mathbb{R}_+^q, \quad w^s, e^s \in \mathbb{R}_+^T, \quad f^s \in \mathbb{R}_+ \quad s = 1, \dots, S, \quad g \in \mathbb{R}_+^n \end{array} \right.$$

Nous allons maintenant prouver que si $\varepsilon < \frac{1}{l_M^{(ST+S+n)}(ST+S+n)^{(ST+S+n)/2}}$, alors toute solution optimale $u' = (x^{2'}, z', y^{1'}, \dots, y^{S'}, w^{1'}, \dots, w^{S'}, e^{1'}, \dots, e^{S'}, f^{1'}, \dots, f^{S'}, g')$, de $D'_\varepsilon(x^{1*}, \lambda')$ vérifie $w^{s'} = 0, s = 1, \dots, S$. Nous pourrions alors conclure la preuve pour PR'_ε .

Réécrivons les contraintes de $D'_\varepsilon(x^{1*}, \lambda')$ en $Hu = v$, où H est la matrice

entière suivante de rang $ST + S + n$,

$$H = \begin{pmatrix} A_2 & B & I_T & -I_T & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ A_2 & 0 & 0 & 0 & 0 & B & I_T & -I_T & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots & & & \\ A_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & B & I_T & -I_T & 0 & 0 \\ \alpha_2 & \beta & 0 & 0 & -1 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ \alpha_2 & 0 & 0 & 0 & 0 & \beta & 0 & 0 & -1 & \cdots & 0 & 0 & 0 & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots & & & \\ \alpha_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & \beta & 0 & 0 & -1 & 0 \\ C_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & -I_n \end{pmatrix},$$

$$u^{tr} = (x^2, y^1, w^1, e_1, f^1, \dots, y^S, w^S, e^S, f^S, g)$$

et

$$v^{tr} = (d^1 - A_1 x^{1*}, \dots, d^S - A_1 x^{1*}, L - \alpha_1 x^{1*}, \dots, L - \alpha_1 x^{1*}, b - C_1 x^{1*}).$$

Remarquons que $\|H\|_\infty = l_M$.

Supposons tout d'abord que u' est une solution de base optimale de $D'_\varepsilon(x^{1*}, \lambda')$. Il existe alors une $(ST + S + n) \times (ST + S + n)$ matrice de base $H_B = (h_{ij})$ et un vecteur de base u'_B tels que

$$u'_B = \frac{1}{\det(H_B)} \text{com}^{tr}(H_B)v.$$

Soit $(t, s), t \in \{1, \dots, T\}$ et $s \in \{1, \dots, S\}$, tel que $w_t^{s'}$ appartienne au vecteur de base, alors il existe τ tel que $w_t^{s'} = \frac{1}{\det(H_B)} (\text{com}^{tr}(H_B)v)_\tau$. En outre, si $(x^{2'}, z', y^1, \dots, y^{S'}, w^1, \dots, w^{S'})$ est une solution optimale de $D'_\varepsilon(x^{1*}, \lambda')$, alors $(x^{1*}, x^{2'}, z', y^1, \dots, y^{S'}, w^1, \dots, w^{S'})$ est une solution optimale de PR'_ε et nous avons d'après (4.7), $w_t^{s'} \leq \varepsilon$. Ainsi $|(\text{com}^{tr}(H_B)v)_\tau| \leq \varepsilon |\det(H_B)|$.

D'après l'inégalité d'Hadamard, nous savons que

$$|(\text{com}^{tr}(H_B)v)_\tau| \leq \varepsilon \|H_B\|_\infty^{(ST+S+n)} (ST + S + n)^{(ST+S+n)/2}.$$

Ainsi si nous fixons $\varepsilon < \frac{1}{\|H_B\|_\infty^{(ST+S+n)} (ST+S+n)^{(ST+S+n)/2}}$, nous avons $|(\text{com}^{tr}(H_B)v)_\tau| < 1$. Or puisque tous les coefficients de H_B et de v sont des entiers, $|(\text{com}^{tr}(H_B)v)_\tau| \in \mathbb{N}$, ainsi $|(\text{com}^{tr}(H_B)v)_\tau| = 0$ et $w_t^{s'} = 0, \forall t = 1, \dots, T$.

Supposons maintenant que u' soit une solution optimale non nécessairement de base de $D'_\varepsilon(x^{1*}, \lambda')$. En utilisant les mêmes arguments qu'à la fin de la preuve de la proposition 6 nous pouvons montrer qu'aucun rayon extrême intervenant dans la décomposition de u' ne peut avoir $w_t^{s'} \neq 0$ pour (s, t) avec $s \in \{1, \dots, S\}$ et

$t \in \{1, \dots, T\}$.

Ainsi toute solution optimale de $D'_\varepsilon(x^{1*}, \lambda')$ vérifie :

$$w^{s'} = 0, \forall s = 1, \dots, S.$$

Pour conclure, considérons une solution optimale $(x^{1*}, x^{2*}, z^*, y^{1*}, \dots, y^{S*}, w^{1*}, \dots, w^{S*})$ de PR'_ε . Il existe λ^* tel que $(\lambda^*, x^{2*}, z^*, y^{1*}, \dots, y^{S*}, w^{1*}, \dots, w^{S*})$ soit une solution optimale de $D'_\varepsilon(x^{1*})$ et que $(x^{2*}, z^*, y^{1*}, \dots, y^{S*}, w^{1*}, \dots, w^{S*})$ soit alors une solution optimale de $D'_\varepsilon(x^{1*}, \lambda^*)$. Ainsi $w^{s*} = 0$ pour tout $s = 1, \dots, S$. La preuve est donc complète. \square

Proposition 8. Soit $\varepsilon < \frac{1}{l_M^{(ST+S+n)}(ST+S+n)^{(ST+S+n)/2}}$. Si (x^*, d^*, y^*, w^*) est une solution optimale de PR_ε alors $w^* = 0$ et (x^*, d^*, y^*) est une solution optimale de PR .

Démonstration. Fixons $\varepsilon < \frac{1}{l_M^{(ST+S+n)}(ST+S+n)^{(ST+S+n)/2}}$. D'après la proposition 7, $w^s = 0$ pour tout $s = 1, \dots, S$ dans toute solution optimale de PR'_ε . Nous en déduisons que $w = 0$ dans toute solution optimale de PR_ε . En effet supposons que ce ne soit pas le cas pour une solution optimale (x^*, d^*, y^*, w^*) de PR_ε ; Il existe alors une solution optimale de PR'_ε avec $x = x^*$ et $w^s > 0$ pour s qui correspond au scénario extrême associé à x^* : c'est une contradiction.

Ainsi, soit (x^*, d^*, y^*, w^*) une solution optimale de PR_ε . Puisque $w^* = 0$, (x^*, d^*, y^*) est une solution de PR ayant $v(PR_\varepsilon)$ pour valeur. Or d'après la proposition 5, $v(PR_\varepsilon) \leq v(PR)$, et (x^*, d^*, y^*) est une solution optimale de PR . \square

Pour résumer nous résolvons PR en résolvant PR_ε : si la solution optimale (x^*, y^*, w^*) de PR_ε vérifie $w^* = 0$ alors (x^*, y^*) est une solution optimale de PR . Sinon $w_t^* > \varepsilon$ pour un certain t , $v(PR_\varepsilon) > U$ et PR n'a pas de solutions. Remarquons que la résolution de PR_ε par l'algorithme 1 ne donne pas accès directement à w^* . Cependant en utilisant le théorème des écarts complémentaires, il est facile à partir d'une solution $(x^*, \lambda^*, \delta^*, \nu^*)$ donnée par l'algorithme 1, de trouver une condition suffisante pour savoir si $w^* = 0$ ou non. En effet (y^*, w^*) est une solution optimale de $R_\varepsilon(x^*, d^*)$ et λ^* est une solution optimale du dual de $R_\varepsilon(x^*, d^*)$:

$$\left| \begin{array}{l} \max_{\lambda} (d^* - Ax^*) \cdot \lambda \\ B^{tr} \lambda \leq \beta \\ \lambda \leq \frac{M}{\varepsilon} \\ \lambda \in \mathbb{R}_+^T \end{array} \right.$$

on a alors, d'après le théorème des écarts complémentaires :

$$w_t \left(\lambda_t - \frac{M}{\varepsilon} \right) = 0, \quad t = 1, \dots, T.$$

Ainsi, si $\lambda_t < M/\varepsilon$ pour tout t , alors $w = 0$ et $(x^*, \lambda^*, \delta^*, \nu^*)$ est aussi une solution optimale de PR .

4.2 Application à un problème de production

Dans cette section, nous testons notre approche sur un problème de production. Une entreprise décide de construire des usines pour fabriquer plusieurs produits. Elle dispose de p usines, l'usine i produisant une quantité a_{it} du produit t , $t = 1, \dots, T$ pour un coût total de production égal à α_i , $i = 1, \dots, p$. La demande d_t concernant le produit t est incertaine et s'écrit $d_t = \bar{d}_t + \delta_t \Delta_t$, pour tout t . Si la production n'est pas suffisante pour satisfaire la demande, l'entreprise a la possibilité d'acheter une certaine quantité, du produit t manquant, à un coût élevé β_t ; cependant, elle ne peut pas acheter une quantité de ce produit supérieure à K_t . Les variables de décisions sont notées x_i , $i = 1, \dots, p$, $x_i = 1$ si l'usine i est construite et $x_i = 0$ sinon. Les variables de recours sont notées y_t , $t = 1, \dots, T$, y_t valant la quantité de produit t que l'entreprise décide d'acheter si la production de ce produit est insuffisante. Nous ne prenons pas en compte dans notre modèle le prix de vente des produits. Le problème à résoudre est le suivant :

$$PR \left\{ \begin{array}{l} \min_x \sum_{i=1}^p \alpha_i x_i + \max_{d \in \mathcal{D}} \min_y \sum_{t=1}^T \beta_t y_t \\ y_t \geq d_t - \sum_{i=1}^p a_{it} x_i \quad \forall t = 1, \dots, T \\ y_t \leq K_t \quad \forall t = 1, \dots, T \\ y \in \mathbb{R}_+^T \\ x_i \in \{0, 1\} \quad i = 1, \dots, p. \end{array} \right.$$

Nous supposons que la demande est toujours satisfaite si les p usines sont construites, c'est-à-dire $\sum_{i=1}^p a_{it} + K_t \geq \bar{d}_t + \Delta_t$ pour tout $t = 1, \dots, T$. Ainsi le problème a toujours une solution. En outre les données sont générées de telle sorte que $a_{it} + K_t < \bar{d}_t$, pour au moins un couple $(i, t) \in \{1, \dots, p\} \times \{1, \dots, T\}$. De cette façon, la propriété de recours complet n'est pas vérifiée. En effet il existe des indices \hat{i} tels que la variable de décision : $x_{\hat{i}} = 1$ et $x_i = 0$ pour tout $i \neq \hat{i}$, il n'y a pas de solution au problème de recours.

Pour s'assurer que $w_t = 0$ pour tout t à la fin de l'algorithme, nous fixons $L = 0$ (tous les coûts étant positifs), $U = \sum_{i=1}^p \alpha_i + \sum_{j=1}^T \beta_j K_j$, correspondant au coût maximum d'une solution dans le pire cas, et $\varepsilon = 1$. En effet ε doit majorer, dans le cas où les variables de décisions sont entières, la valeur absolue du déterminant de n'importe quelle matrice de base de la matrice des contraintes du problème de recours. Cette matrice étant ici totalement unimodulaire, nous n'avons pas besoin d'utiliser l'inégalité d'Hadamard dans l'équation 4.3.

Les données sont générées de la façon suivante : Pour p et T donnés, nous générons de façon aléatoire les coefficients $\alpha, \beta, \bar{d}, K$, entre 0 et 100. Nous générons

ensuite aléatoirement Δ_t entre 0 et 100 pour chaque $t = 1, \dots, T$. Puis nous générons a_{it} de manière aléatoire. Ainsi la condition $a_{it} + K_t < \bar{d}_t$ a de grande chance d'être vérifiée pour au moins un couple $(i, t) \in \{1, \dots, p\} \times \{1, \dots, T\}$. Enfin nous générons $\bar{\delta}_t$ de manière aléatoire entre 0 et T .

Comme nous l'attendions, la résolution du problème par l'algorithme 1 sans les variables artificielles ne donne pas de solution : le programme retourne "Dual infaisable à cause d'une colonne vide". Huit valeurs de (p, T) sont testées. Pour chaque valeur, nous générons cinq instances et nous donnons les résultats moyens dans la table 3.1. La colonne "temps CPU" donne le temps en seconde pour obtenir une solution optimale sur un Bi-pro Intel Nehalem XEON 5570 at 2.93 GHz with 24 Go of RAM. La colonne suivante donne le nombre total d'itérations de l'algorithme 1. La dernière colonne donne le nombre d'itérations où la propriété de recours complet n'est pas vérifiée, c'est à dire le nombre d'itérations où la solution retournée par le problème de recours n'est pas une solution réalisable (c.-à-d. $w > 0$).

p	T	Temps CPU (s)	# itérations	# itérations "non admissible"
100	100	10	32	18
100	1500	754	95	65
300	500	735	87	44
500	300	317	56	39
500	500	2362	117	67
500	800	3480	114	80
600	700	4989	139	98
800	100	118	46	25
1500	100	714	89	41

TABLE 4.1 – Temps CPU moyen, et nombre d'itérations pour p usines et T produits

On remarque que, alors qu'il est facile de résoudre des instances jusqu'à 1500 usines (respectivement produits) et 100 produits (respectivement usines), les instances les plus difficiles sont celles où le nombre d'usines et de produits sont proches. En outre le temps CPU ne semble pas être corrélé au ratio entre le nombre d'itérations et le nombre d'itérations "non admissible".

Finalement, nous remarquons que notre méthode, qui permet de résoudre des problèmes robustes qui ne vérifient pas la propriété de recours complet, est utilisable dans la pratique et permet de résoudre des problèmes avec un grand nombre de variables et de contraintes.

Chapitre 5

Étude d'un problème robuste généralisé à un ensemble d'incertitude quelconque

Dans ce chapitre, nous nous placerons dans le cas où l'ensemble d'incertitude \mathcal{D} est un polytope quelconque dont les points extrêmes ne sont pas connus. Après avoir mis en évidence pourquoi l'approche précédente ne semble pas pouvoir se généraliser directement, nous étudierons une autre approche permettant de résoudre le problème de recours en temps polynomial lorsque le nombre de contraintes est fixé. Pour cela, nous introduirons le zonotope généré par toutes les directions des arêtes de \mathcal{D} . Nous verrons que sous certaines conditions, nous pouvons calculer en temps polynomial tous les points extrêmes de ce zonotope. Nous en déduisons une méthode permettant d'énumérer tous les points extrêmes du polytope \mathcal{D} . Après avoir résolu le problème de recours, nous en déduisons que le problème robuste peut se résoudre en temps polynomial lorsque p et T sont fixés. Dans toute cette section, nous nous référons à [32] pour les résultats concernant la minimisation de fonctions convexes sur des polyèdres.

5.1 Présentation du problème

Dans tout ce chapitre, nous considérons le problème de recours suivant :

$$R \left| \begin{array}{l} \max_{\delta \in \mathcal{D}} \gamma \cdot \delta + \min_y \beta \cdot y \\ By \geq d + D\delta \\ y \in \mathbb{R}_+^q \end{array} \right.$$

où $D \in \mathbb{Q}^{T \times r}$ et \mathcal{D} est un polytope de \mathbb{R}^r et où le terme $-Ax$ est inclus dans d .

Ici le polytope \mathcal{D} est supposé quelconque, comme dans la partie 3.4.2, cependant on ne suppose pas ses points extrêmes connus. Nous permettons en outre aux

scénarios, $\delta \in \mathcal{D}$, d'engendrer un coût, $\gamma.\delta$. On pourrait imaginer une résolution du problème R par un algorithme de génération de contraintes qui calculerait à chaque étape un nouveau point extrême de \mathcal{D} , seulement ce n'est pas l'objet de cette section.

Remarquons aussi que, les points extrêmes de \mathcal{D} étant quelconques, il n'est pas possible de linéariser les produits $\delta_t \lambda_t$, comme nous l'avons fait en 3.2.

Réécrivons le problème de recours $R(x)$, tel qu'il est écrit en 3.2

$$R(x) \left| \begin{array}{l} \max_{d \in \mathcal{D}} \gamma.\delta + \min_y \beta.y \\ By \geq d - Ax \\ y \in \mathbb{R}_+^q. \end{array} \right.$$

et rappelons que l'ensemble \mathcal{D} était défini comme étant l'ensemble

$$\mathcal{D} = \{d : d_t = \bar{d}_t + \delta_t \Delta_t, \sum_{t=1}^T \delta_t \leq \bar{\delta}, 0 \leq \delta_t \leq 1, \forall t = 1, \dots, T\}.$$

Comme l'ensemble des vecteurs δ qui peuvent être des solutions optimales de $R(x)$ sont tels que $\sum_{t=1}^T \delta_t = \bar{\delta}$, on remarque que l'on peut résoudre le problème de recours en résolvant $\binom{T}{\bar{\delta}}$ problèmes linéaires continus. Ainsi lorsque le nombre de contraintes T est fixé, le problème de recours $R(x)$ peut être résolu en temps polynomial. Dans la suite de cette section nous montrerons que ce résultat se généralise au problème R .

Soit f la fonction de \mathcal{D} dans \mathbb{R} telle que

$$f(\delta) = \gamma.\delta + \min_y \beta.y \quad \begin{array}{l} By \geq d + D\delta \\ y \in \mathbb{R}_+^q \end{array}$$

Alors le problème (R) se réécrit :

$$\max_{\delta \in \mathcal{D}} f(\delta).$$

Notons au passage que d'après la proposition 1, la fonction f est convexe.

Nous pouvons donc nous limiter à l'étude des points extrêmes de \mathcal{D} .

5.2 Minimisation d'une fonction convexe et résolution du problème de recours

Avant de voir comment minimiser la fonction f sur le polytope \mathcal{D} , nous devons d'abord donner quelques définitions et propriétés sur les zonotopes et sur les directions des arêtes des polytopes.

Définition 4. Soit \mathcal{P} un polyèdre, et soit a une arête (une face de dimension 1) de \mathcal{P} . Nous dirons que e est une direction de a , si e est un multiple strictement positif de $u - v$, où u et v sont deux points distincts de a .

L'algorithme, présenté dans [32], que nous utiliserons pour minimiser la fonction f , utilise l'ensemble de toutes les directions des arêtes du polytope \mathcal{D} . Avant de présenter cet algorithme, nous devons d'abord donner plusieurs définitions. La définition suivante introduit la notion de cône normale, qui est illustrée à la figure 5.1

Définition 5. Soit \mathcal{P} un polyèdre, le cône normale de \mathcal{P} en un point extrême x^* , $C_{\mathcal{P}}^{x^*}$, est le cône ouvert défini par l'ensemble des vecteurs c tels que x^* soit l'unique solution optimale du programme linéaire $\max_{x \in \mathcal{P}} c \cdot x$.

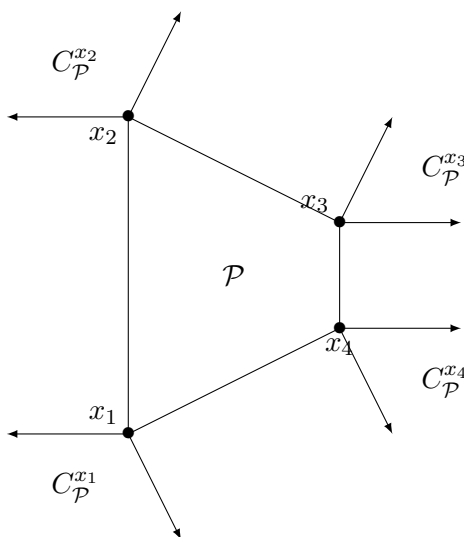


FIGURE 5.1 – Un polyèdre \mathcal{P} et ses cônes normaux

Définition 6. Soit \mathcal{P} un polytope. Nous dirons qu'un polytope Z est un raffinement de \mathcal{P} , si pour tout point extrême z^* de Z , il existe un point extrême x^* de \mathcal{P} tel que le cône de normale de Z en z^* , $C_Z^{z^*}$, soit inclus dans le cône normale de \mathcal{P} en x^* , $C_{\mathcal{P}}^{x^*}$.

Soit $E = \{e_1, \dots, e_m\}$ un ensemble de vecteur. Le zonotope Z , généré par E est défini comme étant :

$$Z = \text{zone}(E) := \text{conv} \left\{ \sum_{i=1}^m \lambda_i e_i, \lambda_i = \pm 1 \right\},$$

où $\text{conv}\{V\}$ représente l'enveloppe convexe de V .

La propriété suivante, due à Minkowski (cf. [20]), permet, en connaissant l'ensemble des directions des arêtes d'un polytope \mathcal{P} , de construire un raffinement de ce polytope.

Proposition 9. Soit \mathcal{P} un polytope et soit E l'ensemble des directions des arêtes de \mathcal{P} .

Alors le zonotope $Z = \text{zone}(E)$, généré par E est un raffinement de \mathcal{P} .

Démonstration. Soit z^* un point extrême de Z , alors il existe des valeurs de $\lambda_e = \pm 1$, telles que $z^* = \sum_{e \in E} \lambda_e e$. Ainsi le cône normal de Z en z^* est égal à l'ensemble des vecteurs c tels que $c \cdot \lambda_e e > 0$ pour tout e . Soit $c \in C_Z^{z^*}$ et soit x^* un point extrême de \mathcal{P} qui est une solution de $\max_{x \in \mathcal{P}} c \cdot x$.

Considérons une arête $[x^*, y]$ de \mathcal{P} , il existe $\bar{e} \in E$ et un réel non nul $\alpha_{\bar{e}}$ tels que $x^* - y = \alpha_{\bar{e}} \bar{e}$. En outre, $0 \leq c \cdot (x^* - y) = c \cdot \alpha_{\bar{e}} \bar{e}$. Or puisque $c \cdot \lambda_e e > 0$ pour tout e , nécessairement $\alpha_{\bar{e}}$ et $\lambda_{\bar{e}}$ ont le même signe et donc $c \cdot \alpha_{\bar{e}} \bar{e} > 0$.

Ainsi, pour tout $c \in C_Z^{z^*}$, $c \cdot (x^* - y) > 0$, pour toute arête de \mathcal{P} contenant x^* , et donc x^* est l'unique solution de $\max_{x \in \mathcal{P}} c \cdot x$. Nous avons donc montré que pour tout

$c \in C_Z^{z^*}$, $c \in C_{\mathcal{P}}^{x^*}$, et donc

$$C_Z^{z^*} \subseteq C_{\mathcal{P}}^{x^*}.$$

□

Exemple 1. Reprenons le polytope \mathcal{P} de la figure 5.1, le zonotope Z engendré par les directions, $E = \{e_1, e_2, e_3\}$, des arêtes de \mathcal{P} est donné à la figure 5.2. On remarque que l'on a bien

$$C_Z^{z_1}, C_Z^{z_6} \subset C_{\mathcal{P}}^{x_1}; C_Z^{z_2} \subset C_{\mathcal{P}}^{x_4}; C_Z^{z_3} \subset C_{\mathcal{P}}^{x_3}; C_Z^{z_4}, C_Z^{z_5} \subset C_{\mathcal{P}}^{x_2}.$$

La proposition suivante (cf. [32, Lemma 2.13]) donne une borne sur le nombre de points extrêmes d'un zonotope.

Proposition 10. Le nombre de points extrêmes d'un zonotope $Z = \text{zone}(E)$ engendré par un ensemble E de m vecteurs de \mathbb{R}^r est en $O(m^{r-1})$. En outre lorsque r est fixé et l'ensemble E est donné, il existe un algorithme polynomial qui construit la liste des points extrêmes, z^* de Z , ainsi que $c_{z^*} \in C_Z^{z^*}$.

Démonstration. Nous supposons que l'ensemble E engendre linéairement \mathbb{R}^r (si non la dimension, r , peut être réduite) et qu'il n'existe pas r points de E qui soient sur le même hyperplan (de dimension $r - 1$). Lorsque E est quelconque la preuve est quasiment la même (voir [24] pour plus de détails).

Soit $H = \{x \in \mathbb{R}^r | c \cdot x = 0\}$, un hyperplan de \mathbb{R}^r , défini par une orientation c . H induit une partition de E en

$$E = H^- \cup H^0 \cup H^+,$$

où $H^- = \{e \in E | c \cdot e < 0\}$, $H^0 = \{e \in E | c \cdot e = 0\}$ et $H^+ = \{e \in E | c \cdot e > 0\}$.

Remarquons alors que les points extrêmes de Z sont en bijection avec les 2-partitions induites par les hyperplans H tels que $H^0 = \emptyset$. En effet lorsque $E = H^- \cup H^+$, le point extrême, $z^* = \sum_{e^+ \in H^+} e^+ - \sum_{e^- \in H^-} e^-$, de Z est l'unique solution de $\max_{z \in Z} c \cdot z$, où c définit l'orientation H .

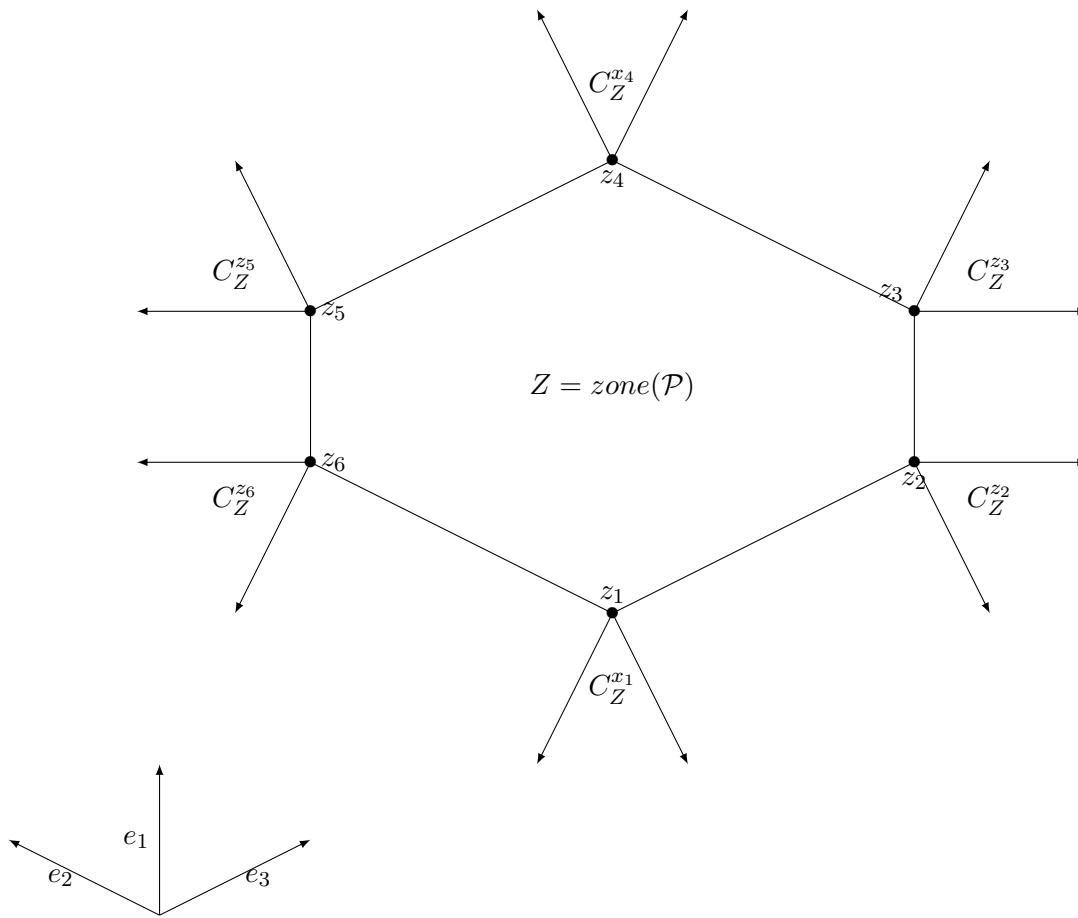


FIGURE 5.2 – Le zonotope $Z = \text{zone}(\mathcal{P})$ et ses cônes normaux

Nous montrons maintenant comment énumérer toutes ces 2-partitions et donc comment énumérer tous les points extrêmes de Z . Soit M l'un des $\binom{m}{r-1}$, sous-ensembles de E de cardinal $r - 1$. Puisqu'il n'existe pas r points de E qui soient sur le même hyperplan, la famille de vecteurs de M est libre et engendre un hyperplan $\hat{H} = \{x \in \mathbb{R}^r | \hat{c}.x = 0\}$. Remarquons que $\hat{H}^0 = M$. En outre, soit L l'un des 2^{r-1} sous-ensembles de M . Puisque M est libre, l'espace engendré par les vecteurs de L n'a que 0 en commun avec l'espace engendré par les vecteurs de $M \setminus L$. Il existe donc un vecteur g qui "sépare" les éléments L des éléments de $M \setminus L$, c'est à dire qu'il existe g tel que $g.x < 0$ pour tout $x \in L$, et $g.x > 0$ pour tout $x \in M \setminus L$. Comme $M = \hat{H}^0 = \{e \in E | \hat{c}.e = 0\}$, il existe ε suffisamment petit tel que l'hyperplan H défini par $c = \hat{c} + \varepsilon g$, ne contienne aucun point de E , et définisse ainsi une 2-partition de E telle que $H^- = \hat{H}^- \cup L$, et $H^+ = \hat{H}^+ \cup (M \setminus L)$. Montrons maintenant que cette procédure permet de construire tous les hyperplans H qui évitent E . En effet, soit H un tel hyperplan. Il existe c tel que $H = \{x \in \mathbb{R}^r | c.x = 0\}$. Soit $e_1 \in E$. On peut construire un

vecteur $c_1 \neq c$ tel que $(c - c_1) \cdot e_1 = 0$. Alors l'hyperplan H_1 défini par le vecteur $c - c_1$ contient au moins un point de E . Supposons que l'on ait pu construire un hyperplan H_p , défini par un vecteur c^p , contenant p points (e_1, \dots, e_p) de E ($p < r - 1$). Soit e_{p+1} un point de E linéairement indépendant de (e_1, \dots, e_p) qui n'appartient pas à H_p . Nous allons montrer que l'on peut perturber H_p en H_{p+1} de façon à ce que $(e_1, \dots, e_p, e_{p+1})$ appartienne à H_{p+1} . Soit $U \in \mathbb{R}^{(p+1) \times r}$ la matrice

dont les lignes sont les e_i , $i = 1, \dots, p + 1$: $U = \begin{pmatrix} e_1 \\ \vdots \\ e_p \\ e_{p+1} \end{pmatrix}$. Alors c_p est solution

de l'équation $Uc = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ e_{p+1} \cdot c_p \end{pmatrix}$, où $e_{p+1} \cdot c_p \neq 0$.

Or le rang de la matrice U étant d'au plus $p + 1 < r$, le nombre de solutions af-

finements indépendantes de $Uc = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ e_{p+1} \cdot c_p \end{pmatrix}$ est d'au moins 2. Il existe donc un

vecteur g non colinéaire à c_p qui soit solution de l'équation.

L'hyperplan H_{p+1} engendré par $c_p - g$ convient alors.

Ainsi quel que soit l'hyperplan H qui évite E , nous pouvons en perturbant H nous ramener à un hyperplan qui touche exactement $r - 1$ points de E . Nous avons donc montré que la méthode décrite plus haut pour engendrer des hyperplans qui évitent E , est exhaustive. Pour finir, remarquons qu'il existe à chaque fois 2 choix possibles pour le vecteur c qui engendre un hyperplan H (c ou $-c$). Ainsi puisqu'il y a $\binom{m}{r-1}$ choix possibles de sous-ensembles M à $r - 1$ éléments, 2^{r-1} sous ensemble de L de M , et deux orientations pour H , on en déduit que le nombre de 2-partitions et donc le nombre de points extrêmes de Z est borné par $2^r \binom{m}{r-1} = O(m^{r-1})$. Remarquons en outre que la méthode décrite plus haut permet d'engendrer les points extrêmes de Z et un vecteur du cône normal en chacun de ces points extrêmes en un temps polynomial. \square

Nous montrons maintenant que si nous connaissons l'ensemble des directions des arêtes, E , de \mathcal{D} , il est possible de résoudre le problème $\max_{\delta \in \mathcal{D}} f(\delta)$ en temps polynomial lorsque la dimension r des vecteurs de \mathcal{D} est fixée. Plus précisément nous montrons comment construire en temps polynomial lorsque r est fixé, un ensemble \mathcal{T} qui contient tous les points extrêmes de \mathcal{D} . La fonction f étant convexe, cela suffit pour maximiser f sur \mathcal{D} en temps polynomial.

En effet, soit r , fixé et soit $Z = \text{zone}(E)$ le zonotope engendré par les directions des arêtes de \mathcal{D} . D'après la proposition 10, on peut construire en temps polynomial la liste des points extrêmes z^s , $s = 1, \dots, S$ de Z ainsi que $c_{z^s} \in$

$C_Z^{z^s}$, $s = 1, \dots, S$.

Pour tout s , soit δ^s une solution optimale de $\max_{\delta \in \mathcal{D}} c_{z^s} \cdot \delta$. Nous démontrons maintenant que pour tout point extrême δ^* de \mathcal{D} , il existe s tel que $\delta^* = \delta^s$.

Soit δ^* un point extrême de \mathcal{D} , et soit $c \in C_{\mathcal{D}}^{\delta^*}$. Considérons un point extrême z^s de Z solution de $\max_{z \in Z} c \cdot z$. Ainsi $c \in C_Z^{z^s}$. D'après la proposition 9, Z est un

raffinement de \mathcal{P} , il existe donc un point extrême $\bar{\delta}$ de \mathcal{D} tel que $C_Z^{z^s} \subseteq C_{\mathcal{D}}^{\bar{\delta}}$. En particulier $c \in C_{\mathcal{D}}^{\bar{\delta}}$. Or par définition des cônes normaux, $C_{\mathcal{D}}^u \cap C_{\mathcal{D}}^v = \emptyset$ pour deux points extrêmes distincts u et v . Nous en déduisons donc que $\bar{\delta} = \delta^*$ et qu'ainsi $C_Z^{z^s} \subseteq C_{\mathcal{D}}^{\delta^*}$.

Nous avons donc que $c_{z^s} \in C_{\mathcal{D}}^{\delta^*}$, et que $\delta^* = \delta^s$.

Ainsi l'ensemble $\mathcal{T} = \{\delta_s, s = 1, \dots, S\}$ contient l'ensemble des points extrêmes de \mathcal{D} .

Pour conclure, nous devons maintenant montrer qu'il est possible de calculer toutes les directions des arêtes de \mathcal{D} en temps polynomial à r fixé.

Dans la définition suivante, nous introduisons la notion de circuit d'une matrice

Définition 7. Soit A une matrice $m \times n$. L'ensemble $\mathcal{C}(A)$, des circuits de A est défini comme étant l'ensemble des vecteurs x de $\mathcal{L}^*(A) = \{x \in \mathbb{Z}^n \mid Ax = 0, x \neq 0\}$, dont le support $\text{supp}(x) = \{x_i \mid x_i \neq 0, i = 1, \dots, n\}$ est minimal dans le sens de l'inclusion et dont les composantes x_i sont premières entre elles.

Remarquons que $c \in \mathcal{A}$ si et seulement si $-c \in \mathcal{A}$.

Le but de la suite de cette partie est de montrer que l'ensemble des circuits d'une matrice correspond à l'ensemble des directions des arêtes d'un polyèdre $\{x \in \mathbb{R}^n \mid Ax = b, x \geq 0\}$. Pour cela nous devons tout d'abord introduire un nouvel ordre partiel sur \mathbb{R}^n . Nous dirons, pour deux vecteurs x, y de \mathbb{R}^n que x est conforme à y , que nous noterons, $x \sqsubseteq y$, si x_i a le même signe que y_i et si $|x_i| \leq |y_i|$, pour tout $i = 1, \dots, n$. En outre nous noterons $x \sqsubset y$ lorsque $x \sqsubseteq y$ et que $x \neq y$. Remarquons que $x \sqsubseteq y$ si et seulement si x et y appartiennent au même orthant de \mathbb{R}^n et si toutes les composantes de x sont inférieures en valeur absolue aux composantes de y .

La proposition suivante permet d'écrire un élément du noyau de A (un x vérifiant $Ax = 0$) comme une somme positive d'éléments du même orthant de \mathbb{R}^n .

Proposition 11. Soit A une matrice $m \times n$ et soit $x \in \mathbb{R}^n$ un vecteur rationnel non nul qui vérifie $Ax = 0$. On peut alors écrire $x = \sum_{i=1}^t \lambda^i y^i$, où $y^i \in \mathcal{A}$, $\lambda^i \geq 0$ et $\lambda^i y^i \sqsubseteq x$ pour tout i .

Démonstration. Soit x un vecteur vérifiant $Ax = 0$. Nous montrons d'abord qu'il existe $c \in \mathcal{C}$ et un réel λ positif tels que $\lambda c \sqsubseteq x$. Supposons que cela soit faux et prenons le contreexemple x ayant le plus petit support. Soit $c \in \mathcal{C}$ tel que $\text{supp}(c) \subsetneq \text{supp}(x)$. Montrons qu'il existe un réel μ tels que $\text{supp}(x - \mu c) \subsetneq \text{supp}(x)$ et que

$x - \mu c$ appartienne au même orthant que x . En effet, en prenant μ suffisamment petit, $x - \mu c$ appartient au même orthant que x . Il suffit alors d'augmenter la valeur de μ jusqu'à ce qu'une composante de $x - \mu c$ s'annule. On a alors le réel μ voulu. Il suffit alors de prendre λ suffisamment petit pour que $\lambda(x - \mu c) \sqsubseteq x$. Or puisque $\text{supp}(\lambda(x - \mu c)) \subsetneq \text{supp}(x)$ et que x était un contre exemple de support minimal, il doit exister $c' \in \mathcal{C}$ et un réel λ' positif tel que $\lambda'c' \sqsubseteq \lambda(x - \mu c)$. C'est une contradiction puisqu'on a alors $\lambda'c' \sqsubseteq \lambda(x - \mu c) \sqsubseteq x$.

Soit donc $c \in \mathcal{C}$ et λ positif tels que $\lambda c \sqsubseteq x$. On peut clairement choisir λ tel que $\text{supp}(x - \lambda c) \subsetneq \text{supp}(x)$. Il suffit alors de poser $y = x - \lambda c \sqsubseteq x$ et de chercher un nouveau $c' \in \mathcal{C}$ et un réel λ' tels que $\lambda'c' \sqsubseteq y$. Le support de y diminuant d'au moins un à chaque étape, on arrive à un moment où $y = 0$ et on a bien écrit x sous la forme $x = \sum_{i=1}^t \lambda^i y^i$, où $y^i \in \mathcal{A}$, $\lambda^i \geq 0$ et $\lambda^i y^i \sqsubseteq x$. \square

Nous pouvons maintenant prouver que l'ensemble des circuits d'une matrice correspond à l'ensemble des directions des arêtes d'un polyèdre défini par cette matrice.

Proposition 12. *Pour toute matrice $m \times n$ A et tout vecteur b , l'ensemble des circuits de A correspond à l'ensemble des directions des arêtes du polyèdre $\mathcal{P} = \{x \in \mathbb{R}^n \mid Ax = b, x \geq 0\}$.*

Démonstration. Soit $e = y - x$ une arête de \mathcal{P} . $Ae = 0$, et il existe d'après la proposition 11 des réels positifs λ^i et des circuits c^i de A tels que $e = \sum_i \lambda^i c^i$, avec $\lambda^i c^i \sqsubseteq e$, pour tout i .

Montrons que $x + \lambda^i c^i \in \mathcal{P}$ pour tout i . En effet $A(x + \lambda^i c^i) = Ax = b$. En outre comme $x \geq 0$, $x + e \geq 0$ et que $\lambda^i c^i \sqsubseteq e$, nous avons $x + \lambda^i c^i \geq 0$. Soit w un vecteur de \mathbb{R}^n tel que les solutions optimales de $\max_{x \in \mathcal{P}} w.x$, soient exactement les points de l'arête e . Nous avons alors que $w.\lambda^i c^i = w.(x + \lambda^i c^i) - w.x \leq 0$, pour tout i , mais $\sum_i w.\lambda^i c^i = w.e = w.y - w.x = 0$. Ainsi $w.\lambda^i c^i = 0$, ce qui prouve que c_i est une direction de l'arête e . \square

Nous pouvons maintenant calculer les ensembles des arêtes du polytope \mathcal{D} . Soit la matrice $\Delta \in \mathbb{R}^{s \times r}$ et le vecteur g tel que $\mathcal{D} = \{\delta \in \mathbb{R}^r \mid \Delta \delta = g\}$, remarquons que si c^1 et c^2 sont deux circuits distincts de Δ , ($c^1 \neq \pm c^2$) alors c^1 et c^2 ne peuvent pas avoir le même support. En effet supposons que ce soit le cas, il existerait alors un réel λ tel que le support de $c^1 - \lambda c^2$ soit strictement inclus dans le support de c^1 et c^2 . En outre par définition des circuits, $c^1 - \lambda c^2 \neq 0$, car deux circuits ne peuvent pas être colinéaires. Contradiction. Nous en déduisons que les circuits de Δ sont entièrement déterminés par leur support. Montrons maintenant que le support d'un circuit de Δ possède au plus s composantes non nulles. En effet soit c un circuit de $\Delta = (\Delta_1, \dots, \Delta_r)$, alors c vérifie

$$\sum_{i \in \text{supp}(c)} c_i \Delta_i = 0.$$

$$\begin{array}{l}
\min_{\substack{x,r \\ y^1, \dots, y^S \\ w^1, \dots, w^S}} \alpha \cdot x + r \\
r \geq \gamma \cdot \delta^s + \beta \cdot y^s, \quad s = 1, \dots, S \\
Ax + By^s \geq d + D\delta^s, \quad s = 1, \dots, S \\
Cx \geq b \\
x_i \in \mathbb{N}, \quad i = 1, \dots, p_1 \quad x_j \in \mathbb{R}_+, \quad j = p_1 + 1, \dots, p \\
y^s \in \mathbb{R}_+^q, \quad s = 1, \dots, S
\end{array}$$

Ainsi, d'après [23] lorsque p_1 , le nombre de variables entières du programme linéaire mixte ci-dessus est fixé, il est possible de trouver une solution optimale du programme en un temps polynomial.

Chapitre 6

Prise en compte des probabilités dans l'approche robuste

Bien que nous ayons maintenant vu comment résoudre le problème robuste PR , il reste des questions, sinon des problèmes, auxquels il convient de répondre. En effet, bien que l'on modélise l'incertitude en mettant de côté toute considération stochastique, est-il possible d'estimer le bien-fondé de nos hypothèses ? S'il s'avère que l'ensemble d'incertitude \mathcal{D} ne correspond pas à la réalité (s'il ne contient pas certains cas très probables par exemple), quelles sont les chances de voir une solution devenir irréalisable ou plus coûteuse ? De plus, si l'on possède des informations sur les données aléatoires d , de quelle façon peut-on s'en servir afin de définir un ensemble \mathcal{D} réaliste, ou au moins plausible ? Enfin, si l'on optimise le coût du pire scénario dans cette approche, qu'en est-il du coût des autres scénarios ? Peut-on, à partir d'une solution et en partant de ce qu'on sait sur d , estimer la probabilité qu'un coût potentiellement différent du "pire" se réalise ? Nous tenterons de répondre à ces questions dans ce chapitre. Très peu de papiers dans la littérature traitent ce problème. Nous pouvons tout de même citer Bertsimas et al. [8], qui étudie les liens entre l'optimisation stochastique et l'optimisation robuste. Dans tout ce chapitre nous supposons sans perte de généralité que l'incertitude ne concerne que le membre droit, d , des contraintes. En outre nous assimilerons celui-ci à un vecteur aléatoire ayant une certaine loi de probabilité.

6.1 Choix du paramètre $\bar{\delta}$

Dans le chapitre 3, nous avons étudié le problème robuste PR en prenant comme ensemble d'incertitude l'ensemble \mathcal{D} défini comme étant :

$$\mathcal{D} = \{d : d_t = \bar{d}_t + \delta_t \Delta_t, \sum_{t=1}^T |\delta_t| \leq \bar{\delta}, -1 \leq \delta_t \leq 1, \forall t = 1, \dots, T\}.$$

Cependant, bien que la valeur de $\bar{\delta}$ joue un rôle décisif sur la valeur d'une solution optimale, nous n'avons pas pour l'instant indiqué quelles valeurs prendre en pra-

tique.

En effet nous avons vu que lorsque $\bar{\delta} = T$, alors le problème robuste est équivalent à l'approche de Soyster (c.f. [34]) qui est très conservatrice et qui donne donc une solution dont le coût est beaucoup plus élevé que la valeur d'une solution du problème déterministe. En revanche, si l'on prend $\bar{\delta} = 0$, alors on est ramené au problème déterministe. Il convient donc de trouver une valeur de $\bar{\delta}$ adéquate donnant un bon équilibre entre le "niveau de protection" de l'ensemble \mathcal{D} , et le coût d'une solution robuste.

Le "niveau de protection" de l'ensemble \mathcal{D} pouvant se définir intuitivement comme étant la proportion de scénarios considérés par rapport au nombre total de scénarios, nous chercherons un ensemble \mathcal{D} tel que, étant donné un scénario d , la probabilité $P(d \in \mathcal{D})$ que d appartienne à \mathcal{D} soit proche de 1.

Soit $\bar{\delta} \in \{0, \dots, T\}$ fixé. Afin d'étudier les cas pour lesquels $P(d \in \mathcal{D})$ est proche de 1, nous allons chercher à minorer $P(d \notin \mathcal{D})$.

Nous supposons que le membre droit des contraintes s'écrit $d_t = \bar{d}_t + \delta_t \Delta_t$, pour tout $t = 1, \dots, T$, où $(\delta_t)_t$ est une famille de variables aléatoires indépendantes de même loi dans $[-1; 1]$, qui admet une densité de probabilité f_δ .

Proposition 13. Soit $\mu = \mathbb{E}(|\delta_1|)$, l'espérance d'un des $|\delta_i|$, et soit $\sigma^2 = \text{Var}(|\delta_1|)$, la variance d'un des $|\delta_i|$. Alors, pour tout $\bar{\delta} \geq T\mu$ Nous avons

$$P(d \notin \mathcal{D}) \leq \exp \left(-T\sigma^2 h \left(\frac{\bar{\delta} - T\mu}{T\sigma^2} \right) \right),$$

où $h(x) = (1+x) \ln(1+x) - x$.

Démonstration. C'est une conséquence directe de l'inégalité de Bennett. En effet posons $X_t = \delta_t - \mathbb{E}(|\delta_t|)$, pour tout $t = 1, \dots, T$. Alors, nous savons d'après l'inégalité de Bennett (c.f. [12]) que pour tout $x \geq 0$,

$$P\left(\sum_{t=1}^T X_t \geq x\right) = \exp \left(-T\sigma^2 h \left(\frac{x}{T\sigma^2} \right) \right).$$

Or, la probabilité $P(d \notin \mathcal{D})$ est égal à $P(\sum_{t=1}^T |\delta_t| > \bar{\delta})$, ce qui achève la preuve. \square

La propriété 13 permet donc, lorsque la loi des δ_t est connue, de calculer numériquement une majoration de $P(d \notin \mathcal{D})$. Nous pouvons alors avoir une idée des valeurs de $\bar{\delta}$ à considérer pour que l'ensemble \mathcal{D} soit pertinent. Nous allons maintenant regarder ce que donne cette borne en pratique lorsque les variables aléatoires δ_t suivent une loi uniforme ou une loi normale tronquée.

6.1.1 Cas d'une loi uniforme

Supposons que les variables aléatoire δ_t suivent une loi uniforme sur $[-1; 1]$, alors $|\delta_t|$ suit une loi uniforme sur $[0; 1]$. Ainsi $\mu = 1/2$, $\sigma^2 = 1/12$, et nous savons d'après le théorème 13 que pour tout $\bar{\delta} \geq T/2$,

$$P(d \notin \mathcal{D}) \leq \exp\left(-\frac{T}{12}h\left(\frac{12\bar{\delta} - 6T}{T}\right)\right). \quad (6.1)$$

Nous donnons maintenant quelques valeurs numériques de cette borne pour différentes valeurs de T et $\bar{\delta}$.

	$\bar{\delta}$			
	55	60	62	65
Valeur de la borne 6.1	0.25	0.008	0.001	$5.9 \cdot 10^{-5}$

TABLE 6.1 – Valeur de la borne 6.1 pour la valeur de $P(d \notin \mathcal{D})$ lorsque $T = 100$

	$\bar{\delta}$			
	260	265	260	285
Valeur de la borne 6.1	0.33	0.09	0.015	$7.2 \cdot 10^{-6}$

TABLE 6.2 – Valeur de la borne 6.1 pour la valeur de $P(d \notin \mathcal{D})$ lorsque $T = 500$

	$\bar{\delta}$			
	510	520	530	550
Valeur de la borne 6.1	0.59	0.1	0.008	$2.9 \cdot 10^{-6}$

TABLE 6.3 – Valeur de la borne 6.1 pour la valeur de $P(d \notin \mathcal{D})$ lorsque $T = 1000$

	$\bar{\delta}$			
	5050	5100	5120	5150
Valeur de la borne 6.1	0.22	0.003	$2.6 \cdot 10^{-4}$	$2.9 \cdot 10^{-6}$

TABLE 6.4 – Valeur de la borne 6.1 pour la valeur de $P(d \notin \mathcal{D})$ lorsque $T = 10000$

6.1.2 Cas d'une loi normale tronquée

Supposons maintenant que les variables aléatoires δ_t suivent une loi normale $N(0, 1)$ tronquée sur $[-1; 1]$. Alors les variables aléatoires $|\delta_t|$ suivent une loi normale $N(0, 1)$ tronquée sur $[0; 1]$, de moyenne $\mu = \frac{f(0)-f(1)}{F(1)-F(0)}$ et de variance $\sigma^2 = 1 - \frac{f(1)}{F(1)-F(0)} - \mu^2$, où f (respectivement F) est la fonction de densité (respectivement la fonction de répartition) de la loi $N(0, 1)$. Nous donnons à nouveau quelques valeurs de la borne 13 pour différentes valeurs de T et $\bar{\delta}$.

	$\bar{\delta}$			
	50	55	57	60
Valeur de la borne 6.1	0.42	0.022	$4.2 \cdot 10^{-3}$	$1.8 \cdot 10^{-6}$

TABLE 6.5 – Valeur de la borne 6.1 pour la valeur de $P(d \notin \mathcal{D})$ lorsque $T = 100$

	$\bar{\delta}$			
	240	250	255	260
Valeur de la borne 6.1	0.30	0.013	$1.3 \cdot 10^{-3}$	$9.6 \cdot 10^{-5}$

TABLE 6.6 – Valeur de la borne 6.1 pour la valeur de $P(d \notin \mathcal{D})$ lorsque $T = 500$

	$\bar{\delta}$			
	510	520	530	550
Valeur de la borne 6.1	0.59	0.1	0.008	$2.9 \cdot 10^{-6}$

TABLE 6.7 – Valeur de la borne 6.1 pour la valeur de $P(d \notin \mathcal{D})$ lorsque $T = 1000$

	$\bar{\delta}$			
	5050	5100	5120	5150
Valeur de la borne 6.1	0.22	0.003	$2.6 \cdot 10^{-4}$	$2.9 \cdot 10^{-6}$

TABLE 6.8 – Valeur de la borne 6.1 pour la valeur de $P(d \notin \mathcal{D})$ lorsque $T = 10000$

6.2 Borne en probabilité sur la valeur du programme de recours

Dans cette section, nous présentons un travail en commun avec Cédric Hervet¹, doctorant à Orange Labs et au CEDRIC.

Nous nous intéressons au problème suivant : étant donné (x^*, d^*, y^*) , une solution optimale du problème robuste PR , quelle est la probabilité, lorsque $x = x^*$, d'avoir une solution optimale ayant un coût supérieur à celui de la solution robuste. En effet, dans la pratique, la valeur optimale y^* des variables de recours y ne pouvant être calculée qu'une fois que l'incertitude est levée, seule la valeur optimale x^* des variables de décision x est pertinente.

En 6.1, nous avons cherché à estimer la probabilité que le second membre d appartienne à l'ensemble d'incertitude \mathcal{D} . Nous pouvons donc à l'aide de cette borne minorer la probabilité que le coût d'une solution ne dépasse pas le coût de la solution robuste. Cependant rien ne dit quelle proportion de scénarios d , parmi ceux n'appartenant pas à \mathcal{D} , induit un coût supérieur à la solution robuste. Dans cette section nous allons donc essayer d'être un peu plus précis, en estimant directement cette probabilité.

6.2.1 Définition et modélisation du problème

Soit (x^*, d^*, y^*) une solution optimale du programme robuste PR et soit Ω un coût fixé (éventuellement égal $v(R(x^*))$), nous cherchons la probabilité suivante :

$$\mathbb{P} = P \left(\min_{\substack{By \geq d - Ax^* \\ y \geq 0}} \beta \cdot y \leq \Omega - \alpha \cdot x^* \right) = P(v(R(x^*, d)) \leq \Omega - \alpha x^*).$$

Notons que la probabilité voulue,

$$P \left(\min_{\substack{By \geq d - Ax^* \\ y \geq 0}} \beta \cdot y > \Omega - \alpha \cdot x^* \right),$$

est alors égale à $1 - \mathbb{P}$. Dans cette section nous supposons que le vecteur aléatoire d est positif (on peut faire cette hypothèse sans perte de généralité puisqu'on peut toujours se ramener à un problème où d est positif).

Bien entendu, calculer directement la valeur de \mathbb{P} semble plutôt ardu. Nous allons maintenant montrer que l'on peut se ramener à un problème équivalent plus simple à résoudre.

1. <http://cedric.cnam.fr/index.php/labo/membre/view?id=298>

Soit \mathfrak{D}_Ω , l'ensemble contenant tous les vecteurs d , tel que $\min_{\substack{By \geq d - Ax^* \\ y \geq 0}} \beta \cdot y \leq \Omega - \alpha \cdot x^*$,

alors $\mathbb{P} = P(d \in \mathfrak{D}_\Omega)$. Nous avons donc transformé notre problème en un autre qui consiste à calculer la probabilité qu'un vecteur aléatoire appartienne à un certain ensemble. Cependant pour que cette approche soit pertinente, nous devons d'abord caractériser l'ensemble \mathfrak{D}_Ω . Dans la proposition suivante, nous montrons que \mathfrak{D}_Ω est un polytope.

Proposition 14. *L'ensemble \mathfrak{D}_Ω est le polytope tel que*

$$\mathfrak{D}_\Omega = \{d \mid \lambda^i \cdot d \leq \Omega - \alpha \cdot x^* + \lambda^i \cdot Ax^*, r^j \cdot d \leq Ax^* \cdot r^j \mid i \in I, j \in J\},$$

où $\{\lambda^i, i \in I\}$ (respectivement $\{r^j, j \in J\}$) est l'ensemble des points extrêmes (respectivement des rayons extrêmes) du polyèdre $\{\lambda \mid B^{tr} \lambda \leq \beta\}$.

Démonstration. En effet soit $d \in \mathfrak{D}_\Omega$, alors en passant au dual de $R(x^*, d)$, nous avons

$$v(R(x^*, d)) = \max_{\substack{B^{tr} \lambda \leq \beta \\ \lambda \geq 0}} (d - Ax^*) \cdot \lambda \leq \mathfrak{D}_\Omega - \alpha \cdot x^*.$$

Ainsi pour tout $\lambda \in \{\lambda \mid B^{tr} \lambda \leq \beta\}$, $(d - Ax^*) \cdot \lambda \leq \mathfrak{D}_\Omega - \alpha \cdot x^*$. En particulier, en considérant les points extrêmes de \mathfrak{D}_Ω , nous avons bien que pour tout $i \in I$,

$$\lambda^i \cdot d \leq \Omega - \alpha \cdot x + \lambda^i \cdot Ax^*.$$

En outre, puisque la valeur optimale du dual est bornée, tout rayon extrême r du polyèdre des contraintes doit vérifier $(d - Ax^*) \cdot r \leq 0$.

Réciproquement, soit $d \notin \mathfrak{D}_\Omega$. Il existe λ^* tel que $(d - Ax^*) \cdot \lambda^* > \mathfrak{D}_\Omega - \alpha \cdot x^*$. Ainsi, si le problème de maximisation est borné, il existe un point extrême qui réalise l'optimum et donc il existe $i \in I$ tel que $\lambda^i \cdot d > \Omega - \alpha \cdot x + \lambda^i \cdot Ax^*$; si le problème de maximisation n'est pas borné, il existe un rayon extrême r tel que $(d - Ax^*) \cdot r > 0$.

Nous avons donc bien prouvé que \mathfrak{D}_Ω était égal à l'ensemble voulu. Pour conclure, remarquons que d , λ^i et r^j étant positifs pour tout i, j , \mathfrak{D}_Ω est bien un polyèdre borné. \square

Remarque 1. *Soit $d \in \mathfrak{D}_\Omega$, alors pour tout $d' \leq d$, $d' \in \mathfrak{D}_\Omega$.*

En effet, pour tout $d' \leq d$, $v(R(x^, d')) \leq v(R(x^*, d)) \leq \Omega - \alpha \cdot x^*$.*

Nous allons maintenant chercher à caractériser \mathcal{H}_D , le plus petit hypercube contenant \mathfrak{D}_Ω .

Proposition 15. *Soit $\mathcal{H}_D = [0; d_1^{max}] \times \dots \times [0; d_T^{max}]$, le plus petit hypercube contenant \mathfrak{D}_Ω . Alors pour tout $t = 1, \dots, T$, d_t^{max} est la valeur à l'optimum du programme linéaire suivant :*

$$P_t \left\{ \begin{array}{l} \max_{d, y} d_t \\ \beta \cdot y \leq \Omega - \alpha \cdot x^* \\ By - d \geq -Ax^* \\ d \geq 0, y \geq 0. \end{array} \right.$$

En outre, \mathcal{H}_D et \mathcal{D}_Ω s'intersectent exactement sur tous les axes de \mathbb{R}^T , c'est à dire que pour tout \bar{t} , le vecteur $d^{\bar{t}}$ tel que $d_t^{\bar{t}} = d_t^{max}$ pour $t = \bar{t}$ et $d_t^{\bar{t}} = 0$ sinon, appartient à $\mathcal{D}_\Omega \cap \mathcal{H}_D$.

Démonstration. En effet, pour tout $t = 1, \dots, T$, d_t^{max} est par définition égal, à la valeur à l'optimum du programme $\max_{\substack{v(R(x^*, d)) \leq \Omega - \alpha \cdot x^* \\ d \geq 0}} d_t$. Ainsi d_t^{max} est égal à la valeur à l'optimum du programme suivant équivalent à P_t :

$$\left| \begin{array}{l} \max_d d_t \\ \min_{\substack{By \leq d - Ax^* \\ y \geq 0}} \beta \cdot y \leq \Omega - \alpha \cdot x^* \\ d \geq 0 \end{array} \right.$$

Montrons maintenant que \mathcal{H}_D et \mathcal{D}_Ω s'intersectent à tous les axes de \mathbb{R}^T . En effet, soit $\bar{t} = 1, \dots, T$ et soit d^* la solution optimale de $P_{\bar{t}}$. Alors le vecteur $d' = (0, \dots, d_{\bar{t}}^*, 0, \dots, 0)$ dont la \bar{t} ème composante est égale à $d_{\bar{t}}^{max}$, appartient à $\mathcal{D}_\Omega \cap \mathcal{H}_D$. En effet, clairement $d' \in \mathcal{H}_D$. En outre, puisque d^* appartient à \mathcal{D}_Ω et que $d' \leq d^*$, d' appartient aussi à \mathcal{D}_Ω , ce qui achève la preuve. \square

La remarque suivante est illustrée à la figure 6.1.

Remarque 2. Remarquons que le ratio entre le volume, $\mathcal{V}(\mathcal{D}_\Omega)$, de \mathcal{D}_Ω et celui de \mathcal{H}_D est borné :

$$\frac{1}{2} \leq \frac{\mathcal{V}(\mathcal{D}_\Omega)}{\mathcal{V}(\mathcal{H}_D)} \leq 1.$$

En effet, \mathcal{D}_Ω et $\mathcal{H}_D = [0; d_1^{max}] \times \dots \times [0; d_T^{max}]$ s'intersectant sur tous les axes de \mathbb{R}^T , nous avons que l'enveloppe convexe constituée de l'origine et des intersections de \mathcal{H}_D avec tous les axes de \mathbb{R}^T (c'est à dire $\text{Conv}\{(0, \dots, 0), (d_1^{max}, 0, \dots, 0), \dots, (0, \dots, 0, d_T^{max})\}$), égale à la demi-pyramide contenant l'origine et "la diagonale" de \mathcal{H}_D , appartient à \mathcal{D}_Ω . Ainsi le volume de \mathcal{D}_Ω est supérieur à la moitié de celui de \mathcal{H}_D .

Exemple 2. Illustrons les résultats que nous avons montré jusqu'à présent sur un exemple. Considérons le problème suivant :

$$R(x^*, d) \left| \begin{array}{l} \min_{y_1, y_2} y_1 + 2y_2 \\ y_1 + y_2 \geq d_1 - x^* \\ y_1 + 3y_2 \geq d_2 - x^* \\ y_1, y_2 \geq 0 \end{array} \right.$$

Nous cherchons donc l'ensemble \mathcal{D}_Ω des vecteurs (d_1, d_2) vérifiant $v(R(x^*, d)) \leq$

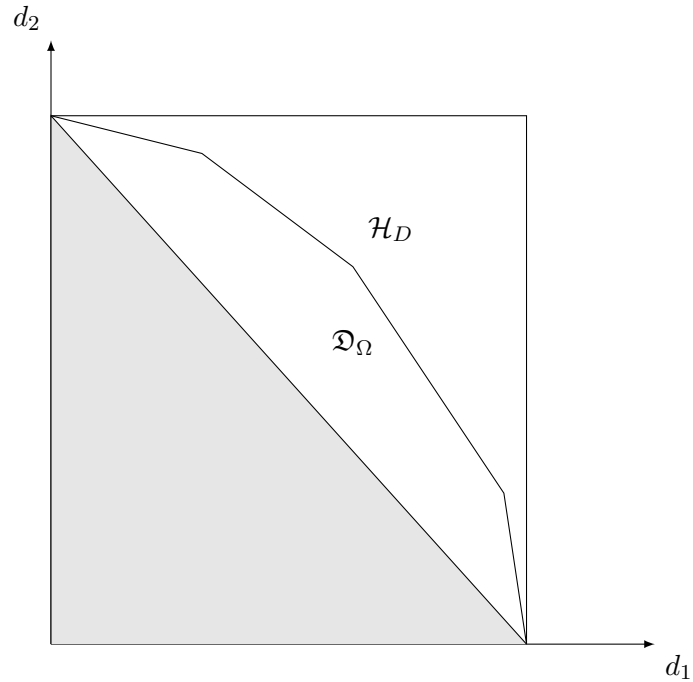


FIGURE 6.1 – Un exemple illustrant la proposition 15 et la remarque 2

$\Omega - x^*$ (ici $\alpha = 1$). Le problème dual s'écrit :

$$R(x^*, d) \left\{ \begin{array}{l} \max_{\lambda_1, \lambda_2} (d_1 - x^*)\lambda_1 + (d_2 - 2x^*)\lambda_2 \\ \lambda_1 + \lambda_2 \leq 1 \\ \lambda_1 + 3\lambda_2 \leq 2 \\ \lambda_1, \lambda_2 \geq 0 \end{array} \right.$$

Et les points extrêmes du polyèdre des contraintes sont $(0, 0)$, $(1, 0)$, $(0, 2/3)$, $(1/2, 1/2)$ (remarquons qu'il n'y a pas de rayons extrêmes).

Nous en déduisons, grâce à la proposition 14, que l'ensemble \mathcal{D}_Ω est défini par :

$$(d_1, d_2) \in \mathcal{D}_\Omega \Leftrightarrow \begin{cases} d_1 \leq \Omega \\ d_2 \leq \frac{3}{2}\Omega + \frac{1}{2}x^* \\ d_1 + d_2 \leq 2\Omega + x^* \\ d_1, d_2 \geq 0 \end{cases}$$

6.2.2 Calcul de \mathbb{P}

Nous pouvons maintenant trouver l'ensemble \mathcal{D}_Ω , cependant, pour avoir une méthode efficace, nous ne pouvons pas nous permettre de calculer tous les points

extrêmes de $\{\lambda \mid B^{tr} \lambda \leq \beta\}$, lorsque la dimension la polyèdre est grande, pour avoir une description complète de \mathcal{D}_Ω . De plus, même si nous l'avions, cela ne nous aiderait pas forcément à calculer la probabilité voulue. Dans le reste de cette section, nous travaillerons avec l'une des deux hypothèses suivante.

Hypothèse 1. *Le vecteur aléatoire d suit une loi uniforme sur un polytope \mathcal{B} dont le volume $\mathcal{V}(\mathcal{B})$ est facile à calculer.*

Pour le calcul de volumes de polytopes, nous renvoyons le lecteur à [14]. Dans la plupart des cas, il s'agit d'un problème difficile.

Hypothèse 2. *Les variables aléatoires d_t , $t = 1, \dots, T$ sont indépendantes et suivent toutes la même loi de probabilité connue.*

Nous donnons maintenant des propositions qui permettent, lorsqu'une des deux hypothèses est vérifiée, de calculer une borne inférieure et supérieure de la probabilité voulue.

Proposition 16. *Sous l'hypothèse 1, si il existe $d' \in \mathcal{D}_\Omega$ et si $\mathcal{D}_\Omega \subseteq \mathcal{B}$, alors*

$$\frac{\prod_{t=1}^T d'_t}{\mathcal{V}(\mathcal{B})} \leq \mathbb{P} \leq \frac{\prod_{t=1}^T d_t^{max}}{\mathcal{V}(\mathcal{B})}.$$

Démonstration. Sous l'hypothèse 1, d suit une loi uniforme. Ainsi, si $\mathcal{D}_\Omega \subseteq \mathcal{B}$, alors $\mathbb{P} = P(d \in \mathcal{D}_\Omega) = \frac{\mathcal{V}(\mathcal{D}_\Omega)}{\mathcal{V}(\mathcal{B})}$.

En outre $\mathcal{V}(\mathcal{D}_\Omega) \leq \mathcal{V}(\mathcal{H}_D) = \prod_{t=1}^T d_t^{max}$, et si $d' \in \mathcal{D}_\Omega$, alors l'hypercube ayant pour extrémités l'origine et d' est inclus dans \mathcal{D}_Ω , et donc $\prod_{t=1}^T d'_t \leq \mathcal{V}(\mathcal{D}_\Omega)$. \square

Proposition 17. *Sous l'hypothèse 2, si il existe $d' \in \mathcal{D}_\Omega$, alors*

$$\prod_{t=1}^T P(0 \leq d_t \leq d'_t) \leq \mathbb{P} \leq \prod_{t=1}^T P(0 \leq d_t \leq d_t^{max}).$$

Démonstration. La preuve est la même que dans la proposition précédente, notons seulement que puisque les variables aléatoires d_t , $t = 1, \dots, T$, sont indépendantes, la probabilité que d appartienne à un hypercube partant de l'origine et d'extrémité d' est égale à $\prod_{t=1}^T P(0 \leq d_t \leq d'_t)$. \square

Remarquons que si la loi de probabilité des d_t est connue, alors nous pouvons calculer le produit $\prod_{t=1}^T P(0 \leq d_t \leq d'_t)$.

Ainsi, sous l'une des deux hypothèses 1 ou 2, nous pouvons alors, connaissant un point d' dans \mathcal{D}_Ω , calculer une borne inférieure de la probabilité voulue. Pour améliorer cette borne, il suffit de considérer d'autres points d dans l'ensemble \mathcal{D}_Ω et de calculer le volume de l'union des hypercubes engendrés par ces points (où la probabilité qu'un vecteur appartienne à une union d'hypercubes).

Nous détaillons ce calcul en sous-section 6.2.3.

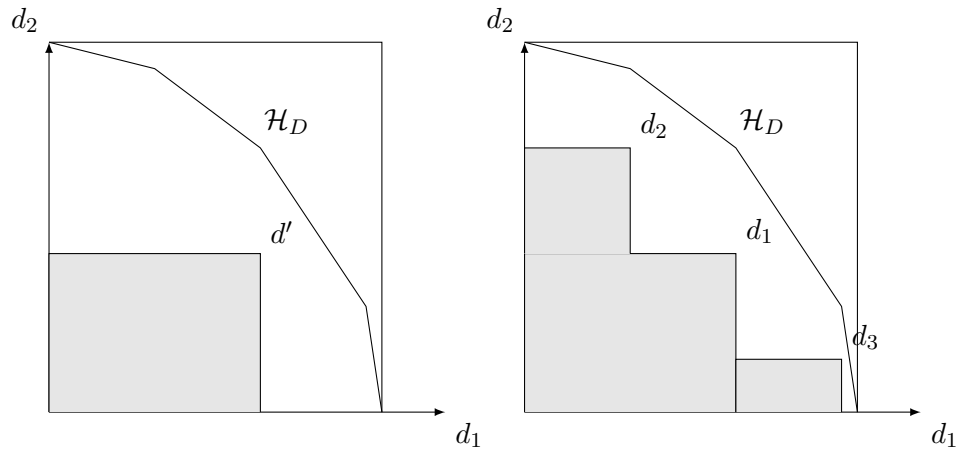


FIGURE 6.2 – Illustration du calcul d’une borne inférieure de \mathbb{P} pour un et plusieurs points dans \mathcal{H}_D .

Notons que nous pouvons appliquer la même méthode pour calculer une borne supérieure de \mathbb{P} . En effet, supposons que nous connaissions un point $d' \in \mathcal{H}_D \setminus \mathcal{D}_\Omega$, et soit H l’hypercube composé des vecteurs d , supérieurs à d' et appartenant à \mathcal{H}_D (voir la figure 6.3). Alors $\mathcal{V}(\mathcal{H}_D) - \mathcal{V}(H)$ est une borne supérieure de \mathbb{P} . On peut ainsi, en considérant plusieurs vecteurs $d' \in \mathcal{H}_D \setminus \mathcal{D}_\Omega$, espérer calculer une borne supérieure, assez proche, de la valeur de \mathbb{P} .

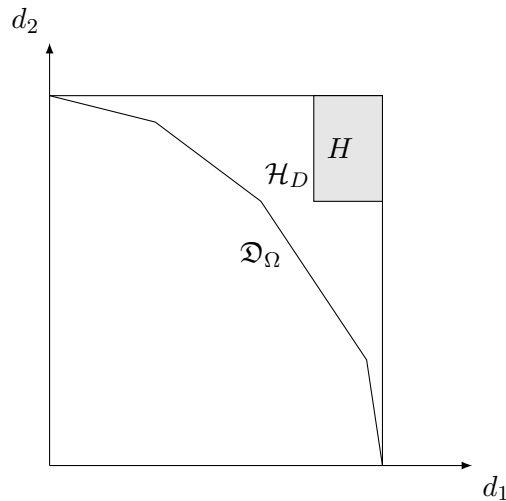


FIGURE 6.3 – Calcul d’une borne supérieure de \mathbb{P}

Remarquons que, d’après la proposition 15, nous pouvons calculer l’hypercube \mathcal{H}_D en un temps polynomial, en résolvant T problèmes linéaires continus. Ainsi,

sous réserve que l'on puisse calculer (c.f. sous-section 6.2.3), le volume d'une union d'hypercube (où la probabilité qu'un vecteur appartienne à cette union), nous disposons d'un algorithme permettant de calculer de manière approchée la probabilité \mathbb{P} .

Cependant, afin de pouvoir mettre en oeuvre notre méthode, nous devons d'abord expliquer comment nous générons nos points d' appartenant à \mathcal{D}_Ω ou à $\mathcal{H}_D \setminus \mathcal{D}_\Omega$. Une méthode serait de générer aléatoirement des vecteurs d' dans \mathcal{H}_D , et, selon que $d' \in \mathcal{D}_\Omega$ ou que $d' \in \mathcal{H}_D \setminus \mathcal{D}_\Omega$, utiliser d' pour le calcul de la borne inférieure ou celui de la borne supérieure. Nous donnons maintenant une méthode qui permet de faire d'une pierre deux coups.

En effet soit $c \in \mathbb{R}_+^T$ un vecteur aléatoire, alors toute solution optimale d^* du programme mathématique $\max_{\substack{v(R(x^*, d)) \leq \Omega - \alpha \cdot x^* \\ d \geq 0}} c \cdot d$, est un point appartenant à la frontière

de \mathcal{D}_Ω et permet donc d'engendrer deux hypercubes, l'un servant pour le calcul de la borne inférieure, l'autre pour le calcul de la borne supérieure. En outre, comme dans la proposition 15, ce programme est équivalent au programme linéaire suivant :

$$P_t \quad \left| \begin{array}{l} \max_{d, y} c \cdot d \\ \beta \cdot y \leq \Omega - \alpha \cdot x^* \\ By - d \geq -Ax^* \\ d \geq 0, y \geq 0. \end{array} \right.$$

On remarque en outre que générer des points d' sur la frontière de \mathcal{D}_Ω , permet d'engendrer des hypercubes qui seront "collés" à la frontière de \mathcal{D}_Ω .

6.2.3 Calcul du volume d'une union d'hypercubes

Nous détaillons maintenant comment calculer le volume d'une union d'hypercubes.

Soit $d = (d_1, \dots, d_T) \in \mathbb{R}_+^T$, on notera $\mathcal{H}(d_1, \dots, d_T)$ l'hypercube ayant l'origine et d pour deux extrémités. En outre, nous noterons $x + \mathcal{H}(d_1, \dots, d_T)$ l'hypercube précédant translaté du vecteur x (ce qui correspond à l'hypercube $[x_1, x_1 + d_1] \times \dots \times [x_T, x_T + d_T]$). Soit $H = x + \mathcal{H}(d_1, \dots, d_T)$ un hypercube, nous noterons $\mathcal{V}(H) = d_1 d_2 \dots d_T$, son volume. Notre approche pour calculer une union d'hypercube est la suivante :

Lorsqu'il n'y a qu'un seul hypercube H^1 dans l'union, alors nous pouvons calculer le volume facilement. Supposons maintenant que l'on veuille calculer le volume de $H^1 \cup H^2$. Pour cela nous allons construire une liste \mathcal{L}^2 d'hypercubes telle que $H^1 \cup H^2 \cup \mathcal{L}^2$, soit un hypercube \mathfrak{H}^2 . Supposons maintenant que pour le calcul d'une union $\bigcup_{k=1}^K H^k$ de K hypercubes, nous connaissions une liste \mathcal{L}^K d'hypercubes telle que que l'union de $\bigcup_{k=1}^K H^k$ avec les hypercubes de la liste \mathcal{L}^K soit un hypercube \mathfrak{H}^K . Alors si l'on considère un nouvel hypercube H^{K+1} , nous verrons

que l'on peut à nouveau, à partir de \mathcal{L}^K et de \mathfrak{H}^K , calculer une nouvelle liste d'hypercubes \mathcal{L}^{K+1} telle que que l'union de ces $K + 1$ hypercubes avec ceux de la liste \mathcal{L}^{K+1} soit un hypercube. En outre, à chaque étape, nous nous assurerons que le volume de l'intersection de deux hypercubes quelconques de la liste soit nul. Muni de \mathfrak{H}^K et de \mathcal{L}^K , nous pourrions calculer le volume d'une union de K hypercubes :

$$\mathcal{V}\left(\bigcup_{k=1}^K H^k\right) = \mathcal{V}(\mathfrak{H}^K) - \sum_{H \in \mathcal{L}^K} \mathcal{V}(H).$$

Nous pourrions calculer de même la probabilité qu'un vecteur appartienne à l'union :

$$P\left(d \in \bigcup_{k=1}^K H^k\right) = \prod_{t=1}^T P(0 \leq d_t \leq \mathfrak{H}_t^K) - \sum_{H \in \mathcal{L}^K} \prod_{t=1}^T P(0 \leq d_t \leq H_t).$$

Pour que cet algorithme fonctionne, nous devons montrer que, étant donné un hypercube $\mathcal{H}(d_1, \dots, d_T)$ et une liste d'hypercubes \mathcal{L} , il est possible de construire pour tout hypercube $H' = \mathcal{H}(d'_1, \dots, d'_n)$, une nouvelle liste d'hypercubes \mathcal{L}' telle que $H \cup H' \cup \mathcal{L}'$ soit un hypercube. Notons que nous devons construire \mathcal{L}' telle que $H \cup H' \cup \mathcal{L}' = \mathcal{H}(\max(d_1, d'_1), \dots, \max(d_n, d'_n))$.

Pour construire \mathcal{L}' , nous procédons en deux étapes. D'abord nous identifions les hypercubes qui sont dans $H' - H$, c'est à dire tous les hypercubes qui "dépassent" de H , lorsque l'on considère l'union de H et H' . Cela nous permet de trouver des hypercubes à rajouter à $H' \cup H$ pour avoir un nouvel hypercube. Ensuite nous devons mettre à jour la liste \mathcal{L} . En effet il se peut que l'intersection de H' avec \mathcal{L} , la liste associée à H , soit non vide.

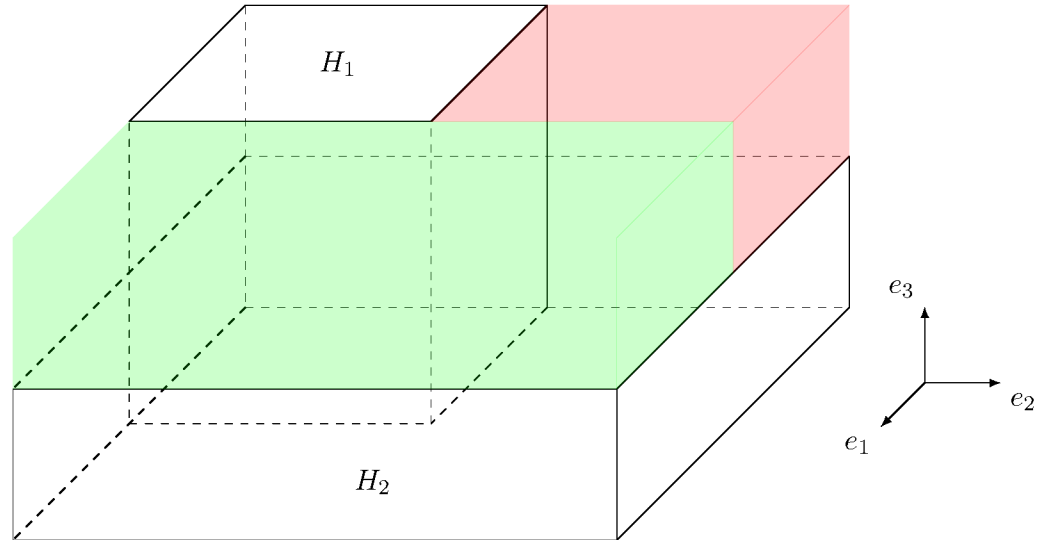


FIGURE 6.4 – Ajout d'hypercubes (en couleur) pour que $H_1 \cup H_2$ soit un hypercube

Supposons que l'on ait une union U de K d'hypercubes partant tous de l'origine et soit $\mathcal{L} = \bigcup_{l=1}^L x^l + \mathcal{H}(d^l)$, un ensemble d'hypercubes qui complète U en

$$\mathfrak{H}^K = \mathcal{H}(d).$$

Soit $H' = \mathcal{H}(d')$, un nouvel hypercube. Quitte à renommer les axes, nous pouvons supposer sans perte de généralité qu'il existe un entier p tel que $d'_i > d_i$, pour tout $i \leq p$ et $d_i \geq d'_i$, sinon ($p = 2$ sur la figure 6.4). Ainsi, nous cherchons un ensemble d'hypercubes \mathcal{L}' qui complète $H \cup H'$ en $\mathfrak{H}^{K+1} = \mathcal{H}(d'_1, \dots, d'_p, d_{p+1}, \dots, d_n)$.

Nous commençons par construire un ensemble d'hypercubes H^i , $i \in \mathcal{I}$, tel que $H \cup H' \cup_{i \in \mathcal{I}} H^i$ soit l'hypercube égal à \mathfrak{H}^{K+1} . Soit i_1 le premier indice i tel que $d'_i < d_i$ ($i_1 = 3$ sur la figure 6.4). Nous supposons sans perte de généralité que $i_1 = p + 1$. Soit l'hypercube

$$H^{p+1} = (0, \dots, d'_{p+1}, \dots, 0) + \mathcal{H}(d'_1, \dots, d'_p, d_{p+1} - d'_{p+1}, d_{p+2}, \dots, d_n).$$

De même, soit i' un autre indice i tel que $d'_i < d_i$, on construit alors l'hypercube

$$H^{i'} = (0, \dots, d'_{i'}, \dots, 0) + \mathcal{H}(d'_1, \dots, d'_{i'-1}, d_{i'} - d'_{i'}, d_{i'+1}, \dots, d_n).$$

On remarque alors que l'ensemble des $H^{i'}$, ainsi construit, complète $H \cup H'$ en un hypercube et que l'intersection de deux H^i quelconques est contenue dans l'hypercube H .

En effet soit x un point de l'hypercube $\mathcal{H}(d'_1, \dots, d'_p, d_{p+1}, \dots, d_n)$ qui n'appartienne pas à $H \cup H'$. Soit i_0 le plus petit indice i pour lequel $x_i > d'_i$, alors $x \in H^{i_0}$. En outre supposons qu'il existe un indice $j_0 > i_0$ tel que $x \in H^{j_0}$. Alors

$$x \in [0, d'_1] \times [0, d'_2] \times \dots \times [d'_{i_0}, d_{i_0}] \times [0, d_{i_0+1}] \times \dots \times [0, d_{j_0}] \times \dots \times [0, d_n] \cap [0, d'_1] \times [0, d'_2] \times \dots \times [0, d'_{i_0}] \times [0, d'_{j_0-1}] \times [d'_{j_0}, d_{j_0}] \times [0, d_{j_0+1}] \times \dots \times [0, d_n]$$

Donc $x_{i_0} = d'_{i_0}$, et x appartient à un ensemble de volume nul. Remarquons cependant que l'intersection d'un de ces hypercubes avec H est de volume non nul.

Soit H^i , $i \in I$, l'ensemble des hypercubes ainsi construit. Afin de construire les hypercubes voulus, nous devons maintenant "retirer" de chaque hypercube H^i , $i \in I$, son intersection avec H . Soit donc $H^i = x + \mathcal{H}(g)$, l'un des hypercubes construit précédemment. Nous allons maintenant construire un ensemble d'hypercubes qui partitionne $H^i - H$.

Soit j_0 le plus petit indice j tel que $x_j + g_j > d_j$, (c'est à dire la première face de H^i qui "sort" de H) alors l'hypercube

$$h_{j_0}^i = x + (0, \dots, d_{j_0} - x_{j_0}, \dots, 0) + \mathcal{H}(\min(g_1, d_1), \dots, \min(g_{j_0-1}, d_{j_0-1}), x_{j_0} + g_{j_0} - d_{j_0}, g_{j_0+1}, \dots, g_n),$$

appartient à $H^i - H$, (Cela correspond à l'hypercube vert sur la figure 1). De même, soit j' un autre indice j pour lequel $x_j + g_j > d_j$, alors l'hypercube

$$h_{j'}^i = x + (0, \dots, d_{j'} - x_{j'}, \dots, 0) + \mathcal{H}(\min(g_1, d_1), \dots, \min(g_{j'-1}, d_{j'-1}), x_{j'} + g_{j'} - d_{j'}, g_{j'+1}, \dots, g_n),$$

appartient à $H^i - H$. En faisant cela pour tous les indices j tels que $x_j + g_j > d_j$, nous construisons un ensemble d'hypercubes qui forment une partition de $H^i - H$. En effet, soit y un point appartenant H^i , mais pas à H , alors il existe un indice \bar{i} tel

Algorithm 2 $Liste(H', H)$

Soit $\mathcal{L} = \emptyset$

pour $t = 1, \dots, T$ **faire**

si $x'_t + d'_t > x_t + d_t$ **alors**

 Construire \bar{H} défini par

$$\begin{aligned} \bar{H} = & (x'_1, \dots, x_t + d_t, x'_{t+1}, \dots, x'_T) + \\ & \mathcal{H}(\min(d'_1, d_1), \dots, \min(d'_{t-1}, d_{t-1}), x'_t + d'_t - x_t - d_t, d'_{t+1}, \dots, d'_T) \end{aligned}$$

$$\mathcal{L} = \mathcal{L} \cup \bar{H}$$

fin si

fin pour

retourner \mathcal{L}

que $y_{\bar{i}} > d_{\bar{i}}$. Prenons le plus petit de ces indices, alors on vérifie que y appartient à $h_{\bar{i}}^i$. En outre en utilisant le même argument que précédemment, on montre que s'il existe un autre indice \bar{j} tel que y appartient aussi à $h_{\bar{i}}^i$, alors y appartient à un ensemble dont le volume est nul.

En appliquant cet algorithme sur tous les hypercubes H^i , on construit une liste d'hypercubes h_j^i , dont l'union deux à deux est de volume nul, qui complète $H \cup H'$ en un hypercube. Pour finir il faut mettre à jour la liste d'hypercubes \mathcal{L} . En effet il faut retirer de tout hypercube H^l appartenant à \mathcal{L} , l'intersection avec H' , et donc appliquer la même procédure que ci-dessus, pour calculer une partition de $H^l - H'$.

Avant de détailler l'algorithme 3 permettant de calculer le volume d'une union d'hypercubes, nous donnons l'algorithme $Liste(H', H)$ qui, étant donné deux hypercubes $H = x + \mathcal{H}(d)$ et $H' = x' + \mathcal{H}(d')$, donne une liste d'hypercubes \mathcal{L} formant une partition de $H' - H$. Nous pouvons maintenant donner en détail, l'algorithme 3 qui, étant donné un hypercube $H = \mathcal{H}(d)$, une liste d'hypercube $\mathcal{L} = \bigcup_{l=1}^L x^l + \mathcal{H}(d^l)$, et un hypercube $H' = \mathcal{H}(d')$, met à jour la liste \mathcal{L} , en une liste \mathcal{L}' telle que $H \cup H' \cup \mathcal{L}'$ soit un hypercube.

Algorithm 3

Calcule de $H' - \mathcal{L}$.

pour $q = 1, \dots, Q$ **faire**

$\bar{\mathcal{L}} = \text{Liste}(H', x^q + \mathcal{H}(d^q))$

$\mathcal{L} = \mathcal{L} \cup \bar{\mathcal{L}} - \{x^q + \mathcal{H}(d^q)\}$

fin pour

On calcule à présent les hypercubes de $H' - H$.

pour $t = 1, \dots, T$ **faire**

si $d'_t < d_t$ **alors**

 Construire \bar{H} défini par

$$\bar{H} = (0, \dots, d'_t, 0, \dots, 0) +$$

$$\mathcal{H}(d'_1, \dots, d'_{t-1}, d_t - d'_t, \max(d'_{t+1}, d_{t+1}), \dots, \max(d'_T, d_T))$$

$\bar{\mathcal{L}} = \text{Liste}(\bar{H}, H)$

$\mathcal{L} = \mathcal{L} \cup \bar{\mathcal{L}}$

fin si

fin pour

Deuxième partie

Dimensionnement robuste d'un système hybride de production d'électricité

Chapitre 7

Présentation du problème

7.1 Introduction

Le développement des énergies renouvelables a créé de nouveaux problèmes d'optimisation combinatoire complexes, comme la conception de systèmes autonomes hybrides de production électrique. Ces systèmes, constitués de plusieurs sources de production d'énergie (vent, soleil, biomasse, ...) sont présentés, entre autre, dans [5, 21, 33]. Dans de nombreux articles, les auteurs cherchent à optimiser la conception d'un tel système et/ou à déterminer son fonctionnement optimal. Les dernières références sur ce problème incluent [15], [16], [22], [31], [33].

Dans [15], Diaf et al. proposent une méthode basée sur la simulation, pour obtenir le nombre optimal d'équipements d'un système hybride comportant des éoliennes et des panneaux solaires. Dans [16], Ferrer-Marti et al., les auteurs modélisent le problème de conception d'un parc hybride éoliennes - panneaux solaires à l'aide d'un programme linéaire mixte en nombres entiers. Cependant ils supposent que les données du problème sont parfaitement connues. Dans [22], Hertz et al. utilisent la programmation linéaire en nombres entiers pour optimiser un réseau d'éoliennes, sans néanmoins considérer l'incertitude sur les données. Dans [31], le but est de déterminer les types, le nombre et l'emplacement d'éoliennes afin d'optimiser à la fois l'énergie produite et les coûts d'installation et de fonctionnement. Leur problème est posé sous la forme d'un programme mathématique qui est ensuite résolu à l'aide de simulations. Dans [33], un système hybride constitué d'éoliennes, de panneaux solaires et de batteries est étudié. Prasad et Natarjan proposent une heuristique pour trouver le nombre d'équipements à installer.

Dans cette partie, nous chercherons à optimiser un parc de production électrique constitué d'éoliennes, de panneaux solaires, de batteries et d'un générateur à diesel, destiné à répondre à une demande locale d'énergie électrique. Ces parcs sont souvent installés dans des régions isolées, comme par exemples des îles ou des petites villes ([9, 16, 28, 30]). Il s'agit de déterminer le nombre d'éoliennes, de panneaux solaires et de batteries à installer afin de répondre à la demande pour un

coût minimum. L'utilisation du générateur à diesel permet de toujours satisfaire la demande, toutefois son utilisation est très onéreuse. Les équipements sont placés sur des sites différents (ensoleillés pour les panneaux solaires, venteux pour les éoliennes) et on connaît le nombre maximal d'équipements de chaque type que l'on peut installer : ce nombre est obtenu à partir de la surface et de la topographie des terrains alloués au parc.

Afin de prendre en compte l'influence des variations saisonnières sur la durée quotidienne d'ensoleillement, la force du vent et la demande, l'étude porte sur une année. L'hypothèse implicite est donc celle d'un fonctionnement globalement similaire chaque année. L'année est divisée en périodes élémentaires pour lesquelles la demande et les conditions climatiques sont connues avec plus ou moins d'incertitude.

Décrivons plus précisément le fonctionnement du parc. Au cours d'une période, si l'énergie éolienne et solaire est suffisante pour répondre à la demande, l'énergie excédentaire produite permet de charger les batteries. En revanche, si cette énergie est insuffisante on utilise d'abord l'énergie éventuellement stockée dans les batteries puis le générateur à diesel si nécessaire. Nous vérifierons par la suite que, sous certaines hypothèses sur les coûts, un tel fonctionnement est optimal.

Au cours d'une période, une batterie peut soit charger de l'énergie, soit restituer de l'énergie chargée précédemment, soit être inactive. Chaque batterie a une capacité maximale, et nécessite d'avoir en stock une quantité minimale d'énergie pour fonctionner ; la charge et la décharge d'énergie, c'est à dire la quantité d'énergie chargée dans les batteries ou déchargée des batteries, par période sont également limitées. De plus, le fonctionnement d'une batterie induit une perte d'énergie : pour une quantité e d'énergie affectée à la charge de la batterie celle-ci n'en restitue qu'une quantité γe ; le coefficient γ , strictement inférieur à 1 est appelé coefficient de rendement. En outre, si les batteries sont pleines et qu'il y a surproduction d'électricité, l'énergie en surplus est utilisée artificiellement et perdue.

L'étude portant sur un an, les coûts des équipements (éoliennes, panneaux solaires, batteries) sont des coûts annuels qui intègrent l'amortissement du matériel (achat et installation) et son entretien (incluant la location éventuel du terrain). L'achat et l'entretien du générateur à diesel induisent des coûts fixes n'intervenant pas dans l'optimisation. En revanche, son fonctionnement implique un coût proportionnel à la quantité d'énergie produite.

Dans un premier temps nous allons étudier le problème en supposant que toutes les données sont parfaitement connues. Cependant cette hypothèse n'est pas très réaliste car il est très difficile de les connaître avec certitude pour chaque période élémentaire : la production d'une éolienne dépend de la force et de la direction du vent, celle d'un panneau solaire de l'ensoleillement et la demande peut être liée à la température ou à d'autres paramètres extérieurs. Nous allons donc, dans un second temps, proposer des modèles prenant en compte cette incertitude. Dans ce cas, le processus de décision se décompose naturellement en deux phases auxquelles sont associés deux ensembles disjoints de variables. La première phase consiste à déterminer le nombre d'équipements de chaque type à installer avant de connaître le

scénario qui se réalisera effectivement. La deuxième phase consiste à déterminer l'utilisation des batteries et du générateur en fonction du scénario qui se réalise et donc la quantité de fuel nécessaire pour l'année. Pour résoudre le problème robuste, nous appliquerons directement la méthode développée dans la partie I, en supposant dans un premier temps que seule la demande est incertaine.

Dans ce chapitre, après avoir modélisé notre problème à l'aide d'un programme linéaire mixte en nombres entiers, nous étudierons la complexité de ce programme : nous montrerons que lorsque le nombre de sources de production d'énergie n'est pas fixé, le problème est NP-difficile. Nous nous attaquerons alors au problème robuste que nous résoudrons directement avec l'approche développée en I. Nous validerons ensuite cette approche par des tests réalisés sur des instances réelles obtenues dans [4].

Dans le chapitre 8, nous montrerons que le programme de recours associé au problème robuste peut être résolu en un temps polynomial, en utilisant un algorithme de programmation dynamique. Nous comparerons ensuite cette approche avec celle du chapitre 7 et nous verrons que lorsque l'incertitude se limite au membre droit des contraintes, la résolution du programme de recours par l'algorithme de programmation dynamique permet de gagner un temps de calcul considérable.

Enfin, dans le chapitre 9, nous généraliserons notre approche, notamment en enrichissant l'ensemble d'incertitude \mathcal{D} .

7.2 Résolution du problème

7.2.1 Modélisation du problème déterministe

Nous faisons pour l'instant abstraction des incertitudes pour construire un modèle mathématique appelé modèle nominal. Nous écrivons tout d'abord un modèle dans lequel les batteries ont toutes un fonctionnement propre, puis nous montrerons que ce modèle est équivalent à un modèle simplifié dans lequel toutes les batteries ont un fonctionnement commun.

Les données du modèle sont :

Données

- T : nombre de périodes $t : [t - 1, t], t = 1, \dots, T$.
- C^w : coût d'installation et d'entretien d'une éolienne sur l'horizon de gestion.
- C^s : coût d'installation et d'entretien d'un panneau solaire sur l'horizon de gestion.
- C^b : coût d'installation et d'entretien d'une batterie sur l'horizon de gestion.
- γ : rendement d'une batterie c'est à dire, proportion de l'énergie stockée dans la batterie, qui pourra effectivement être utilisée.
- K : capacité maximale d'une batterie (en kW.h).

- K_{min} : stock minimal d'une batterie (en kW.h).
- E^{out} : énergie maximale que l'on peut décharger à partir d'une batterie au cours de la période t , $t = 1, \dots, T$. (en kW.h).
- E^{in} : énergie maximale que l'on peut charger dans une batterie au cours de la période t , $t = 1, \dots, T$. (en kW.h).
- E_0^{bat} : énergie présente initialement dans une batterie.
- E_t^w : énergie produite par une éolienne pendant la période t (en kW.h), $t = 1, \dots, T$.
- E_t^s : énergie produite par un panneau solaire pendant la période t (en kW.h), $t = 1, \dots, T$.
- \bar{D}_t : demande au cours de la période t (en kW.h), $t = 1, \dots, T$.
- C^g : coût d'un kW.h d'énergie produite par le générateur à diesel.
- N_{max}^w : nombre maximum d'éoliennes.
- N_{max}^s : nombre maximum de panneaux solaires.
- N_{max}^b : nombre maximum de batteries.

Nous définissons deux types de variables.

Variables de décision d'investissement

- x^w : nombre d'éoliennes à installer dans le parc ($x^w \in \mathbb{N}$).
- x^s : nombre de panneaux solaires à installer dans le parc ($x^s \in \mathbb{N}$).
- x^b : nombre de batteries à installer dans le parc ($x^b \in \mathbb{N}$).

Variables d'exploitation

- $e_{\beta t}^{in}$: énergie chargée dans la batterie β au cours de la période t ($e_{\beta t}^{in} \in \mathbb{R}_+$), $t = 1, \dots, T$, $\beta = 1, \dots, x^b$.
- $e_{\beta t}^{out}$: énergie déchargée de la batterie β au cours de la période t ($e_{\beta t}^{out} \in \mathbb{R}_+$), $t = 1, \dots, T$, $\beta = 1, \dots, x^b$.
- e_t^g : quantité d'énergie produite par le générateur au cours de la période t ($e_t^g \in \mathbb{R}_+$), $t = 1, \dots, T$.
- $e_{\beta t}^b$: énergie présente dans la batterie β à l'instant t , $t = 1, \dots, T$, $\beta = 1, \dots, x^b$.

De plus, nous notons :

- $(e_{\beta}^{in}), (e_{\beta}^{out}), e^g$: les vecteurs définis respectivement par $(e_{\beta,t}^{in})_{t=1,\dots,T}$, $(e_{\beta,t}^{out})_{t=1,\dots,T}$ et $(e_t^g)_{t=1,\dots,T}$, $\beta = 1, \dots, x^b$
- $(e_{\beta,t}) = (e_{\beta,t}^{in}, e_{\beta,t}^{out}, e_t^g)$, $t = 1, \dots, T$, $\beta = 1, \dots, x^b$
- $(e_{\beta}) = (e_{\beta,t})_{t=1,\dots,T}$, $\beta = 1, \dots, x^b$
- $e = (e_{\beta})$, $\beta = 1, \dots, x^b$
- $x = (x^b, x^s, x^w)$.

Fonction coût

La fonction coût C à minimiser s'écrit alors :

$$C = C^s x^s + C^w x^w + C^b x^b + C^g \sum_{t=1}^T e_t^g. \quad (7.1)$$

En effet, le coût du système s'écrit comme la somme du coût d'achat et d'installation des éoliennes, des panneaux solaires, des batteries, et du coût de l'énergie auxiliaire fournie par le générateur à diesel.

Les contraintes se divisent en trois groupes : les contraintes relatives à la demande, celles concernant les batteries et enfin celles qui portent sur la superficie du parc.

Contraintes de demande

Pour que la demande soit satisfaite, l'énergie produite doit être supérieure à \bar{D}_t , au cours de chaque période t .

Au cours de chaque période, les panneaux solaires, les éoliennes et le générateur à diesel produisent une quantité $E_t^s x^s + E_t^w x^w + e_t^g$ d'énergie. En outre pour chaque batterie β , on décharge une quantité $\gamma e_{\beta t}^{out}$ d'énergie, et on charge une quantité $e_{\beta t}^{in}$ d'énergie. On a donc les contraintes :

$$E_t^s x^s + E_t^w x^w + e_t^g + \sum_{\beta=1}^{x^b} (-e_{\beta t}^{in} + \gamma e_{\beta t}^{out}) \geq \bar{D}_t, \quad t = 1, \dots, T \quad (1)$$

Contraintes sur les batteries

— Au cours d'une période t , chaque batterie peut charger une quantité maximale E^{in} d'énergie, et libérer une quantité maximale E^{out} d'énergie, nous avons donc :

$$0 \leq e_{\beta t}^{in} \leq E^{in}, \quad t = 1, \dots, T, \quad \beta = 1, \dots, x^b \quad (2)$$

$$0 \leq e_{\beta t}^{out} \leq E^{out}, \quad t = 1, \dots, T, \quad \beta = 1, \dots, x^b \quad (3)$$

— Le stock dans une batterie à la fin de la période t est égale à l'énergie présente à la fin de la période $t - 1$, moins l'énergie déchargée de la batterie au cours de la période t plus l'énergie stockée au cours de la période t , c'est à dire :

$$e_{\beta t}^b = e_{\beta(t-1)}^b - e_{\beta t}^{out} + e_{\beta t}^{in}, \quad t = 1, \dots, T, \quad \beta = 1, \dots, x^b \quad (4)$$

— Une batterie ne pouvant être simultanément en charge et en décharge, nous supposons qu'au cours d'une même période elle est soit en charge, soit en décharge, soit inactive. Nous avons donc :

$$e_{\beta t}^{in} e_{\beta t}^{out} = 0, \quad t = 1, \dots, T, \quad \beta = 1, \dots, x^b.$$

— En outre chaque batterie est limitée par un stock minimal et une capacité,

$$K_{min} \leq e_{\beta t}^b \leq K, \beta = 1, \dots, x^b, t = 1, \dots, T.$$

Cependant pour une batterie β , on remarque que la dernière contrainte est équivalente à :

$0 \leq e_{\beta t}^b - K_{min} \leq K - K_{min}, \beta = 1, \dots, x^b$. Ainsi en faisant un changement de variable, nous considérerons dans la suite que toutes les batteries ont un stock minimal égale à 0 et une capacité égale à $K - K_{min}$ que nous renommerons K .

Dans un premier temps, nous négligerons la contrainte impliquant qu'une batterie ne peut être en charge et en décharge au cours d'une période. Nous verrons par la suite que notre modèle prend cette contrainte en compte de façon implicite.

Contraintes sur le nombre d'équipements : le parc étant de dimension finie les nombres d'éoliennes et de panneaux solaires que l'on peut installer sont bornés. Nous limitons également le nombre de batteries que l'on peut installer.

$$— x^s \leq N_{max}^s \quad (6)$$

$$— x^w \leq N_{max}^w \quad (7)$$

$$— x^b \leq N_{max}^b \quad (8)$$

On obtient une formulation mathématique de notre problème :

$$\begin{array}{l}
 \min_{x,e} C^s x^s + C^w x^w + C^b x^b + C^g \sum_{t=1}^T e_t^g \\
 \left. \begin{array}{l}
 E_t^s x^s + E_t^w x^w + e_t^g + \sum_{\beta=1}^{x^b} (-e_{\beta t}^{in} + \gamma e_{\beta t}^{out}) \geq \bar{D}_t, \quad t = 1, \dots, T, \quad (1) \\
 0 \leq e_{\beta t}^{in} \leq E^{in}, \quad t = 1, \dots, T, \beta = 1, \dots, x^b \quad (2) \\
 0 \leq e_{\beta t}^{out} \leq E^{out}, \quad t = 1, \dots, T, \beta = 1, \dots, x^b \quad (3) \\
 e_{\beta t}^b = e_{\beta(t-1)}^b - e_{\beta t}^{out} + e_{\beta t}^{in}, \quad t = 1, \dots, T, \beta = 1, \dots, x^b \quad (4) \\
 e_{\beta 0}^b = E_0^{bat} \\
 0 \leq e_{\beta t}^b \leq K, \quad t = 1, \dots, T, \beta = 1, \dots, x^b \quad (5) \\
 x^b \leq N_{max}^b, \quad (6) \\
 x^s \leq N_{max}^s, \quad (7) \\
 x^w \leq N_{max}^w, \quad (8) \\
 x^b, x^w, x^s \in \mathbb{N}, \quad (9) \\
 e_{\beta t}^{in}, e_{\beta t}^{out}, e_{\beta t}^b, e_t^g \in \mathbb{R}_+, \quad t = 1, \dots, T, \beta = 1, \dots, x^b \quad (10)
 \end{array} \right\} P_0
 \end{array}$$

Si P_0 est bien une formulation mathématique du problème, elle est en pratique inutilisable pour le résoudre. En effet P_0 n'est pas écrit sous la forme d'un programme mathématique, le nombre de contraintes de P_0 étant lui même une variable. Pour nous ramener à la résolution d'un programme linéaire, LP défini ci-dessous, nous allons prouver que tout se passe comme si le système ne comportait qu'une seule batterie dont la taille est une variable.

$$\begin{array}{l}
 \min_{x,e} C^s x^s + C^w x^w + C^b x^b + C^g \sum_{t=1}^T e_t^g \\
 E_t^s x^s + E_t^w x^w - e_t^{in} + e_t^g + \gamma e_t^{out} \geq \bar{D}_t, \quad t = 1, \dots, T \quad (1) \\
 0 \leq e_t^{in} \leq x^b E^{in}, \quad t = 1, \dots, T \quad (2) \\
 0 \leq e_t^{out} \leq x^b E^{out}, \quad t = 1, \dots, T \quad (3) \\
 e_t^b = e_{t-1}^b - e_t^{out} + e_t^{in}, \quad t = 1, \dots, T \quad (4) \\
 e_0^b = E_0^{bat} \\
 0 \leq e_t^b \leq x^b K, \quad t = 1, \dots, T \quad (5) \\
 x^b \leq N_{max}^b, \quad (6) \\
 x^s \leq N_{max}^s, \quad (7) \\
 x^w \leq N_{max}^w, \quad (8) \\
 x^b, x^s, x^w \in \mathbb{N}, \quad (9) \\
 e_t^{in}, e_t^{out}, e_t^b, e_t^g \in \mathbb{R}_+, \quad t = 1, \dots, T \quad (10)
 \end{array}$$

Pour cela, nous allons montrer qu'il existe une solution optimale pour laquelle les batteries ont toutes le même comportement à chaque période t . C'est à dire que si une batterie est en charge (respectivement décharge) aucune autre ne peut être en décharge (respectivement charge).

Proposition 18. *Il existe une solution optimale $(x, e_\beta)_{\beta=1, \dots, x^b}$ de P_0 telle que :*

$$e_{\beta t}^{in} \cdot e_{\beta t}^{out} = 0, \quad t = 1, \dots, T, \quad \beta = 1, \dots, x^b.$$

Démonstration. Soit un vecteur $(x, e_\beta)_{\beta=1, \dots, x^b}$ de P_0 optimal.

Supposons qu'il existe τ et β_1 tel que $e_{\beta_1 \tau}^{in} > 0$ et $e_{\beta_1 \tau}^{out} > 0$. Alors, il existe $\epsilon > 0$ tel que $e_{\beta_1 \tau}^{in} - \epsilon > 0$ et $e_{\beta_1 \tau}^{out} - \epsilon > 0$.

On définit le vecteur e^ϵ par :

$$e_{\beta_1 t}^\epsilon = e_{\beta_1 t}, \quad \forall t < \tau, \quad \forall t > \tau, \quad e_{\beta_1 \tau}^{\epsilon, in} = e_{\beta_1 \tau}^{in} - \epsilon, \quad e_{\beta_1 \tau}^{\epsilon, out} = e_{\beta_1 \tau}^{out} - \epsilon, \quad e_\tau^{\epsilon, aux} = \max(e_\tau^g - \epsilon(1-\gamma), 0),$$

et $e_\beta^\epsilon = e_\beta$ pour tout $\beta \neq \beta_1$.

Alors $(e^\epsilon)_{\beta=1, \dots, x^b}$ est un vecteur admissible car :

— si $\max(e_\tau^g - \epsilon(1 - \gamma), 0) = e_\tau^g - \epsilon(1 - \gamma)$, alors

$$\begin{aligned} e_\tau^{\epsilon,aux} + \sum_{\beta=1}^{x^b} (-e_{\beta\tau}^{\epsilon,in} + \gamma e_{\beta\tau}^{\epsilon,out}) &= e_\tau^g - \epsilon(1 - \gamma) + \sum_{\beta=1}^{x^b} (-e_{\beta\tau}^{in} + \gamma e_{\beta\tau}^{out}) + \epsilon(1 - \gamma) \\ &= e_\tau^g + \sum_{\beta=1}^{x^b} (-e_{\beta\tau}^{in} + \gamma e_{\beta\tau}^{out}) \geq \bar{D}_\tau, \end{aligned}$$

— si $\max(e_\tau^g - \epsilon(1 - \gamma), 0) = 0$, alors

$$\begin{aligned} e_\tau^{\epsilon,aux} + \sum_{\beta=1}^{x^b} (-e_{\beta\tau}^{\epsilon,in} + \gamma e_{\beta\tau}^{\epsilon,out}) &= 0 + \sum_{\beta=1}^{x^b} (-e_{\beta\tau}^{in} + \gamma e_{\beta\tau}^{out}) + \epsilon(1 - \gamma) \\ &\geq \sum_{\beta=1}^{x^b} (-e_{\beta\tau}^{in} + \gamma e_{\beta\tau}^{out}) + e_\tau^g \geq \bar{D}_\tau. \end{aligned}$$

Donc dans tous les cas,

$$e_\tau^{\epsilon,aux} + \sum_{\beta=1}^{x^b} (-e_{\beta\tau}^{\epsilon,in} + \gamma e_{\beta\tau}^{\epsilon,out}) \geq e_\tau^g + \sum_{\beta=1}^{x^b} (-e_{\beta\tau}^{in} + \gamma e_{\beta\tau}^{out}) \geq \bar{D}_\tau.$$

De plus, le stock de la batterie à la fin de la période τ est inchangé donc les autres contraintes sont inchangées.

En outre $e_\tau^{\epsilon,aux} \leq e_\tau^g$. La solution $(e^\epsilon)_\beta$ est donc au moins aussi bonne que $(e)_\beta$. \square

Cette propriété implique non seulement qu'une batterie donnée n'est pas simultanément en charge et en décharge, mais encore qu'aucune batterie n'est en charge si une autre est en décharge. En effet si ce n'est pas le cas on voit que l'on peut refaire exactement le même raisonnement en considérant deux batteries β_1, β_2 telles que : $e_{\beta_1\tau}^{in} > 0$ et $e_{\beta_2\tau}^{out} > 0$. Nous noterons qu'en pratique, il se trouve que cela correspond avec le fonctionnement réel d'un ensemble de batteries.

Dans la suite nous pouvons donc supposer que tout se passe comme si le système comportait une seule batterie dont les caractéristiques sont sa capacité $x^b K$, son rendement γ , sa charge maximale par période $x^b E^{in}$, sa décharge maximale par période $x^b E^{out}$, et son prix $x^b C^b$.

P_0 est donc équivalent au programme linéaire LP réécrit ci-dessous.

$$\begin{array}{l}
 LP \left\{ \begin{array}{l}
 \min_{x,e} C^s x^s + C^w x^w + C^b x^b + C^g \sum_{t=1}^T e_t^g \\
 E_t^s x^s + E_t^w x^w - e_t^{in} + e_t^g + \gamma e_t^{out} \geq \bar{D}_t, \quad t = 1, \dots, T \quad (1) \\
 0 \leq e_t^{in} \leq x^b E^{in}, \quad t = 1, \dots, T \quad (2) \\
 0 \leq e_t^{out} \leq x^b E^{out}, \quad t = 1, \dots, T \quad (3) \\
 e_t^b = e_{t-1}^b - e_t^{out} + e_t^{in}, \quad t = 1, \dots, T \quad (4) \\
 e_0^b = E_0^{bat} \\
 0 \leq e_t^b \leq x^b K, \quad t = 1, \dots, T \quad (5) \\
 x^b \leq N_{max}^b, \quad (6) \\
 x^s \leq N_{max}^s, \quad (7) \\
 x^w \leq N_{max}^w, \quad (8) \\
 x^b, x^s, x^w \in \mathbb{N}, \quad (9) \\
 e_t^{in}, e_t^{out}, e_t^b, e_t^g \in \mathbb{R}_+, \quad t = 1, \dots, T \quad (10)
 \end{array} \right.
 \end{array}$$

où les variables e_t^{in}, e_t^{out} représentent respectivement l'énergie totale, chargée dans les batteries et déchargée des batteries pendant la période t , $t = 1, \dots, T$, et où les variables e_t^b représentent le stock total d'énergie présent dans les batteries à l'instant t .

On note \mathcal{P}^e l'ensemble défini par les contraintes (2), (3), (4), (5), (10), relatives aux variables d'exploitation, et \mathcal{P}^x , l'ensemble défini par (6), (7), (8), (9), relatives aux variables de décision. LP se réécrit :

$$LP \left\{ \begin{array}{l}
 \min_{x,e} C^s x^s + C^w x^w + C^b x^b + C^g \sum_{t=1}^T e_t^g \\
 E_t^s x^s + E_t^w x^w - e_t^{in} + e_t^g + \gamma e_t^{out} \geq \bar{D}_t, \quad t = 1, \dots, T, \quad (1) \\
 (e^{in}, e^{out}, e^b) \in \mathcal{P}^e, \quad (11) \\
 x \in \mathcal{P}^x
 \end{array} \right.$$

7.2.2 Comportement optimal de la batterie et du générateur

Nous allons montrer maintenant que si x^b, x^s, x^w ont des valeurs fixées, des valeurs optimales des variables d'exploitation (e^{in}, e^{out}, e^g) se déduisent simplement à partir du problème d'optimisation LP . Nous montrons que, à coût C^g constant (ou décroissant), le comportement du système décrit en section 7.1 est optimal. C'est à dire que : en cas de surproduction d'énergie, on charge ce qu'on peut dans la batterie ; en cas de sous-production d'énergie, on utilise d'abord ce qui est stocké dans la batterie pour répondre au manque avant d'utiliser le générateur. Soit

$x^s = \tilde{x}^s, x^w = \tilde{x}^w, x^b = \tilde{x}^b$ fixés et soit $\tilde{E}_t = E_t^s \tilde{x}^s + E_t^w \tilde{x}^w$, l'énergie solaire et éolienne fournie au cours de la période t .

Proposition 19. Une solution optimale de $LP(\tilde{x}^b, \tilde{x}^s, \tilde{x}^w)$ est obtenue en fixant e de la façon suivante :

Pour tout $t = 1, \dots, T$:

— si $\tilde{E}_t \geq \bar{D}_t$ alors

$$\begin{aligned} e_t^{in} &= \min(\tilde{E}_t - \bar{D}_t, \tilde{x}^b E^{in}, \tilde{x}^b K - e_{t-1}^b) \\ e_t^{out} &= 0 \\ e_t^g &= 0. \end{aligned}$$

— si $\tilde{E}_t < \bar{D}_t$ alors

$$\begin{aligned} e_t^{in} &= 0 \\ e_t^{out} &= \min\left(\frac{\bar{D}_t - \tilde{E}_t}{\gamma}, \tilde{x}^b E^{out}, e_{t-1}^b\right), \\ e_t^g &= \bar{D}_t - \tilde{E}_t - \gamma e_t^{out}. \end{aligned}$$

Démonstration. Pour deux vecteurs u et v , nous noterons ici que $u < v$ si et seulement si $u \leq v$ et si il existe au moins un indice i tel que $u_i < v_i$.

Notons \tilde{e} le vecteur e défini ci-dessus, on voit que \tilde{e} est un vecteur admissible de LP , qui correspond au fonctionnement décrit dans la section 7.1. En effet, lorsque $\tilde{E}_t \geq \bar{D}_t$, alors la quantité d'énergie que l'on peut charger dans la batterie pendant la période t est limitée par : la quantité d'énergie en surplus, la charge maximale de la batterie au cours d'une période, et la capacité de la batterie. Ce qui correspond bien à l'expression de e_t^{in} donné. Lorsque $\tilde{E}_t < \bar{D}_t$ nous avons de même que la quantité d'énergie que l'on peut décharger de la batterie pendant la période t est limitée par : la quantité nécessaire pour combler le manque : $\frac{\bar{D}_t - \tilde{E}_t}{\gamma}$ (puisque l'on décharge en fait γe_t^{out}), la quantité maximale d'énergie que l'on peut décharger de la batterie au cours d'une période, et le stock d'énergie dans batterie au temps $t-1$.

Soit e quelconque admissible pour $LP(\tilde{x}^b, \tilde{x}^s, \tilde{x}^w)$.

- Considérons les instants t où $\tilde{E}_t \geq \bar{D}_t$ alors, l'excès d'énergie produite par les éoliennes et les panneaux solaires est utilisé pour charger le maximum d'énergie possible dans la batterie. En effet, si ce n'est pas le cas, une partie de cette énergie est perdue, ce qui ne peut pas donner une meilleure solution.
- on considère maintenant les instants t où $\tilde{E}_t < \bar{D}_t$, Le vecteur e solution doit vérifier $\tilde{E}_t - e_t^{in} + \gamma e_t^{out} + e_t^g = \bar{D}_t$. Nous avons déjà montré que $e_t^{in} e_t^{out} = 0$ pour tout t . Montrons qu'une solution \bar{e} qui vérifie $\bar{e}^{out} < \tilde{e}^{out}$, n'est jamais optimale, c'est à dire qu'il n'est jamais intéressant d'utiliser le

générateur si la batterie n'est pas vide (car le coût est constant ou décroissant).

Soit τ , la première période telle que $\bar{e}_\tau^{out} < \tilde{e}_\tau^{out} = \min\left(\frac{\bar{D}_\tau - \tilde{E}_\tau}{\gamma}, \tilde{x}^b E^{out}, e_{\tau-1}^b\right)$.

Le stock de la batterie à la date $\tau + 1$ est alors

$$\bar{e}_{\tau+1}^b = \tilde{e}_{\tau+1}^b + (\tilde{e}_\tau^{out} - \bar{e}_\tau^{out}).$$

Dans le meilleur des cas, la batterie pourra utiliser ce surplus d'énergie en entier au cours des périodes $t \geq \tau + 1$, ce qui se traduira par un gain en énergie auxiliaire de $-\gamma(\tilde{e}_\tau^{out} - \bar{e}_\tau^{out})$, au cours des périodes $t \geq \tau + 1$: $\sum_{t=\tau+1}^T \bar{e}_t^g \geq \sum_{t=\tau+1}^T \tilde{e}_t^g - \gamma(\tilde{e}_\tau^{out} - \bar{e}_\tau^{out})$.

Cependant au cours de la période τ , $\bar{e}_\tau^{out} < \tilde{e}_\tau^{out}$ implique que

$$\bar{e}_\tau^{aux} = \tilde{e}_\tau^{aux} + \gamma(\tilde{e}_\tau^{out} - \bar{e}_\tau^{out}).$$

Étant donné que le coût en énergie auxiliaire C^g est constant, on prouve que la solution $e = \bar{e}$ est dans le meilleur des cas équivalente à la solution $e = \tilde{e}$. □

Nous avons donc montré que lorsque x^b, x^s, x^w ont des valeurs fixées, il était possible de trouver le vecteur (e) optimal en temps linéaire.

7.2.3 Complexité

Dans le problème précédent, modélisé par le programme LP , il y a deux sources d'énergie renouvelable ainsi qu'une batterie et un générateur. Il est facile de généraliser ce modèle au cas où il y a non plus deux, mais n sources d'énergie renouvelable. Il s'agit alors de trouver le nombre optimal d'équipement de chaque source d'énergie à installer pour répondre à la demande au moindre coût. Nous notons LP_{gen} le programme mathématique généralisé qui en résulte. Nous supposons qu'un élément de la source i , coûte C^i et produit une quantité E_t^i d'énergie pendant la période t , $t = 1, \dots, T$. La fonction objectif devient $\sum_{i=1}^n C^i x^i + C^b x^b + C^g \sum_{t=1}^T e_t^g$. Les contraintes (1) s'écrivent $\sum_{i=1}^n E_t^i x^i - e_t^{in} + \gamma e_t^{out} + e_t^g \geq \bar{D}_t, t = 1, \dots, T$ et les contraintes (6) – (8) sont remplacées par $x^i \leq N_{max}^i, i = 1, \dots, n$. Les contraintes (2) – (5), (9) et (10) sont inchangées.

Nous prouvons maintenant que LP_{gen} est NP-difficile. Il n'existe donc pas d'algorithme polynomial permettant de résoudre LP_{gen} , à moins que $P = NP$.

Proposition 20. LP_{gen} appartient à la classe des problèmes NP-difficiles.

Démonstration. Nous montrons que le problème de sac à dos borné BKP se réduit à LP_{gen} . Soit $DBKP$, le problème de décision associé à BKP :

$$(DBKP) \begin{cases} \text{Donné : } n, a_1, a_2, \dots, a_{n+1}, c_1, c_2, \dots, c_{n+1}, u_1, u_2, \dots, u_{n+1}, b, V \text{ dans } \mathbb{N} \\ \text{Question : Existe-t-il } (y_1, \dots, y_{n+1}) \in \mathbb{N}^{n+1} \text{ tel que} \\ \sum_{i=1}^{n+1} a^i y^i \geq V, \sum_{i=1}^{n+1} c^i y^i \leq b \text{ et } y_i \leq u_i, i = 1, \dots, n+1. \end{cases}$$

Nous définissons maintenant DLP_{gen} , le problème de décision associé à LP_{gen} :

$$(DLP_{gen}) \begin{cases} \text{Donné : } n, T, W, C^b, C^g, E^{in}, E^{out}, e_0^b, \gamma, K, N_{max}^b, \bar{D}_t, C^i, E_t^i, N_{max}^i, \\ t = 1, \dots, T, i = 1, \dots, n. \\ \text{Question : Existe-t-il un vecteur } (x^1, \dots, x^n) \in \mathbb{N}^n, \text{ un entier } x^b, \text{ et} \\ \text{des réels positifs } e_t^{in}, e_t^{out}, e_t^b, e_t^g, t = 1, \dots, T, \text{ vérifiant les} \\ \text{contraintes de } LP_{gen} \text{ et tels que la valeur de la fonction objectif} \\ \text{soit inférieure ou égale à } W? \end{cases}$$

A partir d'une instance de $DBKP$, nous construisons une instance de DLP_{gen} : $T = 1$; il y a n sources d'énergie, $C^i = c_i$, $E_1^i = a_i$ et $N_{max}^i = u_i$ pour $i = 1, \dots, n$; $C^b = c_{n+1}$, $K = a_{n+1}$, $N_{max}^b = u_{n+1}$, $E^{in} = E^{out} = e_0^b = a_{n+1}$, $\gamma = 1$; $C^g = 2b$; $\bar{D}_1 = V$; $W = b$. Remarquons que nous pouvons alors fixer $e_1^{in} = 0$ et $e_1^{out} = a_{n+1}x^b$, ce qui implique que $e_1^b = 0$, d'après la contrainte (4) (En effet le problème n'ayant qu'une seule période, on peut toujours, sans perte de généralité, décharger la batterie au maximum pendant la première et unique période).

Nous devons maintenant prouver qu'il existe une solution à $DBKP$ si et seulement si il existe $(x^1, \dots, x^n) \in \mathbb{N}^n$, $x^b \in \mathbb{N}$, e_1^{in} , e_1^{out} , e_1^b , $e_1^g \in \mathbb{R}_+$ tels que les contraintes suivantes soient vérifiées :

$$\begin{cases} \sum_{i=1}^n c_i x^i + c_{n+1} x^b + 2be_1^g \leq b \quad (**) \\ \sum_{i=1}^n a_i x^i + a_{n+1} x^b + e_1^g \geq V \quad (*) \\ x^i \leq u_i, i = 1, \dots, n \\ x^b \leq u_{n+1} \\ x^b, x^i (i = 1, \dots, n) \in \mathbb{N}, e_1^g \in \mathbb{R}_+ . \end{cases}$$

Soit y_1, y_2, \dots, y_{n+1} une solution de $DBKP$. Clairement, en posant $x^i = y_i$ pour tout $i = 1, \dots, n$, $x^b = y_{n+1}$ et $e_1^g = 0$, nous obtenons une solution de DLP_{gen} .

Réciproquement, soit $((x^i)_{1 \leq i \leq n}, x^b, e_1^g)$ une solution de DLP_{gen} , i.e. de l'ensemble de contraintes ci-dessus. D'après la contrainte (*), $\sum_{i=1}^n a_i x^i + a_{n+1} x^b \geq V - e_1^g$ et, d'après (**), $e_1^g \leq 1/2$; ainsi, nous avons $\sum_{i=1}^n a_i x^i + a_{n+1} x^b \geq V - 1/2$. Or, puisque $\sum_{i=1}^n a_i x^i + a_{n+1} x^b$ et V sont des entiers, nous obtenons : $\sum_{i=1}^n a_i x^i + a_{n+1} x^b \geq V$. Ainsi, le vecteur y défini par $y_i = x^i$ pour $i = 1, \dots, n$,

$y_{n+1} = x^b$ est une solution de $DBKP$, et les deux problèmes sont bien équivalents.

Puisque $DBKP$ est NP-complet (c.f. [19]), nous avons bien prouvé que le problème de décision, DLP_{gen} , associé au problème généralisé LP_{gen} , est aussi NP-complet. \square

7.3 Résolution du problème robuste

Supposons maintenant que certaines données du problème soient incertaines. Par souci de clarté, nous supposons d'abord, comme dans la partie I, que l'incertitude ne concerne que la demande à chaque période, D_t , qui peut varier dans un ensemble borné \mathfrak{D} . Rappelons que notre objectif robuste est de trouver une solution réalisable (x, e) qui minimise le coût total induit par le pire scénario D associé à x . Le problème robuste s'écrit :

$$PR \left\{ \begin{array}{l} \min_{x \in \mathcal{P}^x} C^s x^s + C^w x^w + C^b x^b \\ + \max_{D \in \mathfrak{D}} \min_{e \in \mathcal{P}^e} C^g \sum_{t=1}^T e_t^g \\ E_t^s x^s + E_t^w x^w - e_t^{in} + \gamma e_t^{out} + e_t^g \geq D_t \quad t = 1, \dots, T \end{array} \right.$$

Nous supposons à partir de maintenant que le stock initial d'énergie dans la batterie est nul.

Les variables x sont les variables de décision et les variables e sont celles de recours. Pour tout x réalisable, le programme de recours $R(x)$ s'écrit :

$$R(x) \left\{ \begin{array}{l} \max_{d \in \mathfrak{D}} \min_{e \in \mathcal{P}^e} C^g \sum_{t=1}^T e_t^g \\ E_t^s x^s + E_t^w x^w - e_t^{in} + \gamma e_t^{out} + e_t^g \geq D_t \quad t = 1, \dots, T \end{array} \right.$$

Comme dans la partie I, nous définissons :

$$\mathfrak{D} = \{D \in \mathbb{R}_+^T : D_t = \bar{D}_t + \delta_t \Delta_t, \sum_{t=1}^T \delta_t \leq \bar{\delta}, 0 \leq \delta_t \leq 1, \forall t = 1, \dots, T\},$$

où \bar{D}_t, Δ_t et $\bar{\delta}$ sont donnés. Ainsi, D_t peut varier entre \bar{D}_t et $\bar{D}_t + \Delta_t$, pour $t = 1, \dots, T$. Δ_t est égal à la variation maximale de D_t , δ_t représente l'incertitude sur D_t , et $\bar{\delta}$ borne la somme totale des variations de D . Si $\bar{\delta} \geq T$, alors quelle que soit la valeur de x , le scénario extrême associé sera $D_t = \bar{D}_t + \Delta_t$ pour tout t ; si $\bar{\delta} = 0$ alors il n'y a qu'un seul scénario : $D_t = \bar{D}_t$ pour tout t ; ainsi $\bar{\delta}$ est à choisir entre 0 et T . Nous renvoyons le lecteur à la section 6.1 pour plus de précisions sur le choix de $\bar{\delta}$.

Soit $x \in \mathcal{P}^x$ et $D \in \mathcal{D}$, nous définissons, de la même façon qu'en section 3.1.1, le programme linéaire suivant :

$$\begin{array}{l}
 R(x, D) \left| \begin{array}{l}
 \min_{e \in \mathcal{P}^e} C^g \sum_{t=1}^T e_t^g \\
 - e_t^{in} + \gamma e_t^{out} + e_t^g \geq D_t - E_t^w x^w - E_t^s x^s, \quad t = 1..T \quad (1) \\
 0 \leq e_t^{in} \leq x^b E^{in}, \quad t = 1, \dots, T \quad (2) \\
 0 \leq e_t^{out} \leq x^b E^{out}, \quad t = 1, \dots, T \quad (3) \\
 e_t^b = e_{t-1}^b - e_t^{out} + e_t^{in}, \quad t = 1, \dots, T \quad (4) \\
 e_0^b = 0 \\
 0 \leq e_t^b \leq x^b K, \quad t = 1, \dots, T \quad (5) \\
 e_t^{in}, e_t^{out}, e_t^b, e_t^g \in \mathbb{R}_+, \quad t = 1, \dots, T \quad (10)
 \end{array} \right.
 \end{array}$$

dont le dual, $DR(x, D)$ s'écrit :

$$\begin{array}{l}
 DR(x) \left| \begin{array}{l}
 \max_{\alpha, \beta, \lambda, \mu, \pi} \sum_{t=1}^T [(D_t - E_t) \lambda_t - x^b E^{in} \alpha_t - x^b E^{out} \beta_t - x^b K \pi_t] \\
 \text{s.c. } \lambda_t \leq C^g, \quad t = 1, \dots, T \quad (13) \\
 - \lambda_t - \alpha_t + \mu_t \leq 0, \quad t = 1, \dots, T \quad (14) \\
 \gamma \lambda_t - \beta_t - \mu_t \leq 0, \quad t = 1, \dots, T \quad (15) \\
 \mu_{t+1} - \mu_t - \pi_t \leq 0, \quad t = 1, \dots, T \quad (16) \\
 \alpha_t, \beta_t, \lambda_t, \mu_t, \pi_t \geq 0 \quad t = 1, \dots, T \quad (17)
 \end{array} \right.
 \end{array}$$

Soient les variables duales, $\lambda, \alpha, \beta, \mu, \pi$ associées respectivement aux contraintes (1), (2), (3), (4), (5). Nous suivons alors l'approche développée dans la section 3.2, c'est à dire, dualiser $R(x, D)$ afin de transformer $R(x)$ en un problème quadratique $DR(x)$ tel que $v(DR(x)) = v(R(x))$:

$$\begin{array}{l}
 DR(x) \left| \begin{array}{l}
 \max_{\alpha, \beta, \delta, \lambda, \mu, \pi} \sum_{t=1}^T [(\bar{D}_t + \delta_t \Delta_t - E_t) \lambda_t - x^b E^{in} \alpha_t - x^b E^{out} \beta_t - x^b K \pi_t] \\
 \text{s.c. } \lambda_t \leq C^g, \quad t = 1, \dots, T \quad (13) \\
 - \lambda_t - \alpha_t + \mu_t \leq 0, \quad t = 1, \dots, T \quad (14) \\
 \gamma \lambda_t - \beta_t - \mu_t \leq 0, \quad t = 1, \dots, T \quad (15) \\
 \mu_{t+1} - \mu_t - \pi_t \leq 0, \quad t = 1, \dots, T \quad (16) \\
 \sum_{t=1}^T \delta_t \leq \bar{\delta} \quad (18) \\
 \delta_t \in \{0, 1\}, \quad t = 1, \dots, T \quad (19) \\
 \alpha_t, \beta_t, \lambda_t, \mu_t, \pi_t \geq 0 \quad t = 1, \dots, T \quad (17)
 \end{array} \right.
 \end{array}$$

Où $E_t = E_t^s x^s + E_t^w x^w$ et où selon la proposition 2, $0 \leq \delta_t \leq 1$ est remplacé par $\delta_t \in \{0, 1\}$. Nous linéarisons ensuite $DR(x)$ en remplaçant les produits $\delta_t \lambda_t$ par les variables ν_t . Nous obtenons alors le programme linéaire mixte suivant :

$$\begin{array}{l}
 \max_{\alpha, \beta, \delta, \lambda, \mu, \nu, \pi} \sum_{t=1}^T [(\bar{D}_t - E_t) \lambda_t - x^b E^{in} \alpha_t - x^b E^{out} \beta_t - x^b K \pi_t + \Delta_t \nu_t] \\
 \text{s.c. } \lambda_t \leq C^g, \quad t = 1, \dots, T \quad (13) \\
 -\lambda_t - \alpha_t + \mu_t \leq 0, \quad t = 1, \dots, T \quad (14) \\
 \gamma \lambda_t - \beta_t - \mu_t \leq 0, \quad t = 1, \dots, T \quad (15) \\
 \mu_{t+1} - \mu_t - \pi_t \leq 0, \quad t = 1, \dots, T \quad (16) \\
 \sum_{t=1}^T \delta_t \leq \bar{\delta} \quad (18) \\
 \delta_t \in \{0, 1\}, \quad t = 1, \dots, T \quad (19) \\
 \nu_t \leq C^g \delta_t, \quad t = 1, \dots, T \quad (20) \\
 \nu_t \leq \lambda_t, \quad t = 1, \dots, T \quad (21) \\
 \alpha, \beta, \lambda, \mu, \nu, \pi \geq 0 \quad (17)
 \end{array}$$

Soit \mathcal{P}_Q le polyèdre défini par les contraintes (13), ..., (21) de $LDR(x)$ où nous avons remplacé (19) par $0 \leq \delta_t \leq 1$, et soit $(\mathcal{P}_Q)_I = \text{conv}(\mathcal{P}_Q \cap \{\delta \in \{0, 1\}^T\})$, l'enveloppe convexe des solutions admissibles de $LDR(x)$. Cette enveloppe convexe ne dépend pas de x et, $(\mathcal{P}_Q)_I$ étant un polyèdre, nous pouvons réécrire le problème robuste PR en le programme linéaire suivant :

$$\begin{array}{l}
 \min_{x, z} C^s x^s + C^w x^w + C^b x^b + z \\
 \text{s.c. } z \geq \sum_{t=1}^T [(\bar{D}_t - E_t^s x^s - E_t^w x^w) \lambda_t^s - x^b E^{in} \alpha_t^s \\
 - x^b E^{out} \beta_t^s - x^b K \pi_t^s + \Delta_t \nu_t^s], \quad s = 1 \dots S. \\
 x^b \leq N_{max}^b, \\
 x^s \leq N_{max}^s, \\
 x^w \leq N_{max}^w, \\
 z \geq 0, \quad x^b, x^w, x^s \in \mathbb{N}
 \end{array}$$

où $S = |\mathcal{S}|$ et $\mathcal{S} = \{(\alpha^s, \beta^s, \lambda^s, \mu^s, \nu^s, \pi^s)_{1 \leq s \leq S}\}$ est l'ensemble des points extrêmes de $(\mathcal{P}_Q)_I$.

Nous pouvons alors résoudre $PROB$ par l'algorithme 4, de génération de contraintes, qui correspondant à l'algorithme 1 de la section 3.2, réécrit pour notre modèle.

Comme dans la section 3.4, nous pouvons généraliser nos résultats au cas où l'énergie, produite par une éolienne ou un panneau solaire à chaque période t , est incertaine. Nous définissons alors les ensembles \mathfrak{E}^w et \mathfrak{E}^s comme :

Algorithm 4 Algorithme de génération de contrainte pour résoudre PR

- 1: $(\alpha^0, \beta^0, \lambda^0, \mu^0, \nu^0, \pi^0) = 0$. Définir $L \leftarrow -\infty, U \leftarrow +\infty, k \leftarrow 1$.
- 2: Résoudre le problème maître :

$$PROB^k \left\{ \begin{array}{l} \min_{x,z} C^s x^s + C^w x^w + C^b x^b + z \\ s.c. z \geq \sum_{t=1}^T [(\bar{D}_t - E_t^s x^s - E_t^w x^w) \lambda_t^s - x^b E^{in} \alpha_t^s \\ \quad - x^b E^{out} \beta_t^s - x^b K \pi_t^s + \Delta_t \nu_t^s], \quad 0 \leq s \leq k-1. \\ x^b \leq N_{max}^b, \\ x^s \leq N_{max}^s, \\ x^w \leq N_{max}^w, \\ z \geq 0, x^b, x^w, x^s \in \mathbb{N}. \end{array} \right.$$

Soit (x^k, z^k) la solution obtenue.

$$L \leftarrow (C^b, C^s, C^w).x^k + z^k.$$

- 3: Résoudre $LDR(x^k)$. Soit $(\lambda^k, \delta^k, \nu^k)$ la solution optimale.

$$U \leftarrow \min\{U, (C^b, C^s, C^w).x^k + v(DR(x^k))\}.$$

si $U = L$, **alors** retourner (x^k, z^k) **sinon** aller à 4.

- 4: Ajouter la contrainte

$$z \geq \sum_{t=1}^T (\bar{d}_t - (Ax)_t) \lambda_t^k + \Delta_t \nu_t^k,$$

au problème maître $PROB^k$, $k \leftarrow k + 1$ et aller en 2.

$$\mathfrak{E}^s = \{E^s : E_t^s = \bar{E}_t^s - \phi_t \Phi_t, \sum_{t=1}^T \phi_t \leq \bar{\phi}, 0 \leq \phi_t \leq 1, \forall t = 1, \dots, T\}.$$

$$\mathfrak{E}^w = \{E^w : E_t^w = \bar{E}_t^w - \omega_t \Omega_t, \sum_{t=1}^T \omega_t \leq \bar{\omega}, 0 \leq \omega_t \leq 1, \forall t = 1, \dots, T\}.$$

7.4 Résultats numériques

Dans cette section, nous appliquons notre méthode pour optimiser un réseau de production d'électricité, isolé dans le Montana, USA. Nous utilisons ici des instances réelles présentées dans [4].

Les tests ont été réalisés sur un Bi-pro. Intel Nehalem XEON 5570 à 2.93 GHz ayant 24 Go de RAM.

Le système est composé d'éoliennes de type BWC XL.1, ayant une puissance nominale de 1.24 kW ; de panneaux solaires de taille 1000 kW et de batteries Trojan L16P. En Table 7.1, nous donnons les principales valeurs des paramètres du système. Le nombre T de périodes est égale 8760, ce qui correspond au nombre d'heures dans une année. Les coûts unitaires, C^w , C^s et C^b sont calculés de la manière suivante : $\frac{\text{coût d'achat}}{\text{durée de vie}} + \text{coût annuel d'entretien}$. Le niveau d'incertitude Δ_t sur la demande à la période t , est égal à dix pour cent de la demande moyenne à la période t . Les bornes sur le nombre d'éoliennes, de panneaux solaires et de batteries sont toutes égales à 500. Les autres données sont directement extraites de [4]. Nous ne rappelons pas non plus les 8760 valeurs moyennes de la demande et de la production d'énergie éolienne et solaire.

Nous étudions deux cas ; dans le premier, l'incertitude ne concerne que la demande. Dans le second, l'incertitude porte aussi sur la production d'énergie éolienne et solaire. Supposons en premier lieu que l'incertitude ne concerne que la demande.

C^w	C^s	C^b	C^g	K	γ	E^{in}	E^{out}	$N_{max}^s, N_{max}^w, N_{max}^b$
295.0\$	280.0\$	26.0\$	3.9\$	2.16	0.85	0.11	2.16	500

TABLE 7.1 – Données principales

Sept valeurs de $\bar{\delta}$ sont testées, les résultats sont donnés en Table 7.2.

Pour chaque valeur de $\bar{\delta}$, nous donnons le nombre optimal d'éoliennes, de panneaux solaires, et d'éléments dans la batterie. Nous donnons aussi la valeur optimale du programme robuste.

	$\bar{\delta} (T = 8760)$						
	0	100	500	700	850	1000	8760
x^w	45	46	48	48	48	48	48
x^s	64	65	67	67	67	67	67
x^b	467	470	499	499	499	499	499
Coût (en \$)	49874.5	51564	54540.3	54901.9	54901.9	54901.9	54901.9

TABLE 7.2 – Nombre d'équipements et coût robuste du parc en fonction du niveau d'incertitude $\bar{\delta}$.

Comme attendu, la valeur optimale du problème robuste est une fonction strictement croissante de $\bar{\delta}$ jusqu'à un certain seuil (ici 700), seuil qui est lui même une fonction croissante de la demande D . En effet, la valeur optimale du programme robuste est bien une fonction croissante du niveau d'incertitude, puisque toute solution du problème, lorsque $\bar{\delta} = \hat{\delta}$, permet aussi de répondre à toute demande

lorsque $\bar{\delta} < \hat{\delta}$. De plus, une fois que la taille du parc est suffisante pour couvrir la demande sur un ensemble de périodes critiques, alors la taille du parc est suffisante pour couvrir la demande sur toutes les périodes. Ainsi le coût reste constant une fois que $\bar{\delta}$ est suffisamment grand pour que cet ensemble de périodes critiques soit couvert par la production d'énergie éolienne et solaire.

Pour les temps de calcul, nous renvoyons le lecteur à la section 8.2

Nous revenons maintenant au cas général, où l'incertitude peut intervenir aussi bien dans la demande que dans la production d'énergie éolienne et solaire. Dans la Table 7.3, nous donnons les résultats de l'algorithme de génération de colonnes, ainsi que le nombre d'itérations, pour différentes valeurs de $(\bar{\delta}, \bar{\phi}, \bar{\omega})$.

	# itérations	x^w	x^s	x^b	Coût
$\bar{\delta} = 0, \bar{\phi} = 500, \bar{\omega} = 0$	27	49	69	476	51844.3
$\bar{\delta} = 0, \bar{\phi} = 0, \bar{\omega} = 500$	27	48	66	465	51348.4
$\bar{\delta} = 8760, \bar{\phi} = 0, \bar{\omega} = 0$	26	49	72	499	54901.9
$\bar{\delta} = 0, \bar{\phi} = 8760, \bar{\omega} = 0$	30	48	71	456	51852.0
$\bar{\delta} = 0, \bar{\phi} = 0, \bar{\omega} = 8760$	29	51	64	469	51438.2
$\bar{\delta} = 0, \bar{\phi} = 8760, \bar{\omega} = 8760$	26	50	71	471	53362.7
$\bar{\delta} = 100, \bar{\phi} = 100, \bar{\omega} = 100$	29	47	67	468	53743
$\bar{\delta} = 500, \bar{\phi} = 500, \bar{\omega} = 500$	28	49	74	500	58182.6

TABLE 7.3 – Nombre d'équipements et coût robuste du parc en fonction des niveaux d'incertitudes $\bar{\delta}$, $\bar{\phi}$ et $\bar{\omega}$

Comme précédemment, la valeur optimale du programme de recours est une fonction strictement croissante de $\bar{\delta}$, $\bar{\phi}$ et $\bar{\omega}$, jusqu'à un certain seuil. De plus, nous remarquons que l'influence combinée des différentes incertitudes sur le coût de la solution robuste est supérieure à l'influence de ces incertitudes, prises séparément. En effet, une fois que les $\bar{\delta}$ périodes critiques concernant l'incertitude sur la demande sont fixées, une augmentation de ϕ et de ω induira un coût supplémentaire sur les $\bar{\delta}$ périodes critiques précédentes.

Chapitre 8

Résolution du problème de recours en temps polynomial

Dans les chapitres précédents, nous avons vu deux méthodes pour résoudre le problème de recours $R(x)$: En section 3.2, le problème de recours est reformulé un programme linéaire mixte avec des variables 0 – 1; en section 5.2, l'optimum du programme de recours est trouvé en résolvant un programme linéaire en chaque point extrême du polyèdre des incertitudes \mathfrak{D} . Cependant, le programme de recours étant NP-difficile (c.f. [29]), il n'existe pas d'algorithme polynomial permettant de résoudre le problème de recours dans le cas général. Dans ce chapitre, nous montrons que le problème de recours associé à notre problème de conception de parc hybride, peut se résoudre en temps polynomial en utilisant un algorithme de programmation dynamique. Après avoir présenté l'algorithme en section 8.1, nous comparerons les temps de résolution du problème robuste, avec ou sans l'algorithme de programmation dynamique.

8.1 L'algorithme de programmation dynamique

Le principe de l'algorithme est le suivant : à chaque étape, nous résolvons un problème de recours "tronqué", $R_x(\tau, \zeta, \beta)$, défini sur les $(T - \tau + 1)$ dernières périodes ($\tau \in \{1, \dots, T\}$), c'est à dire défini de $\tau - 1$ à T . $\beta = e_{\tau-1}^b$, est le stock de la batterie au temps $\tau - 1$, et ζ , ($\zeta \in \mathbb{N}$, $\zeta \leq \bar{\delta}$) borne la somme des variations de la demande sur ces $(T - \tau + 1)$ dernières périodes. Nous pouvons formuler $R_x(\tau, \zeta, \beta)$ en le programme mathématique suivant :

$$\begin{array}{l|l}
R_x(\tau, \zeta, \beta) & \begin{array}{l}
\max_{\substack{\sum_{t=\tau}^T \delta_t \leq \zeta \\ 0 \leq \delta_t \leq 1, t=\tau, \dots, T}} \min_e C^g \sum_{t=\tau}^T e_t^g \\
- e_t^{in} + \gamma e_t^{out} + e_t^g \geq D_t - E_t^s x^s - E_t^w x^w, t = \tau, \dots, T \quad (1) \\
e_t^{in} \leq E^{in}, t = \tau, \dots, T \quad (2) \\
e_t^{out} \leq E^{out}, t = \tau, \dots, T \quad (3) \\
e_t^b \leq K, t = \tau, \dots, T \quad e_{\tau-1}^b = \beta \quad (5) \\
e_t^b = e_{t-1}^b - e_t^{out} + e_t^{in}, t = \tau, \dots, T \quad (4) \\
e_t^{in}, e_t^{out}, e_t^b, e_t^g \in \mathbb{R}_+, t = \tau, \dots, T \quad (10)
\end{array}
\end{array}$$

où nous avons supposé, sans perte de généralité, que $x^b = 1$. Remarquons que (τ, ζ, β) représente "l'état actuel" du processus de décision et que $R(x) = R_x(1, \bar{\delta}, 0)$. Par souci de clarté, nous utilisons les notations suivantes : $\bar{E}_t = \bar{D}_t - E_t^s x^s - E_t^w x^w$, $E_t = D_t - E_t^s x^s - E_t^w x^w = \bar{E}_t + \delta_t \Delta_t$, et $v(\tau, \zeta, \beta) = v(R_x(\tau, \zeta, \beta))$. Soit $f(E_\tau, \beta) = e_\tau^b$, l'énergie chargée dans la batterie au temps τ , et soit $g(E_\tau, \beta) = e_\tau^g$, l'énergie fournie par le générateur durant la période τ , lorsque β est le stock de la batterie au temps $\tau - 1$. D'après la proposition 19, pour tout $t = 1, \dots, T$, nous avons

$$\left. \begin{array}{l}
f(E_t, \beta) = \max \left(\beta - \frac{E_t}{\gamma}, \beta - E^{out}, 0 \right) \\
g(E_t, \beta) = E_t - \gamma \min \left(\beta, E^{out}, \frac{E_t}{\gamma} \right)
\end{array} \right\} \text{ si } E_t \geq 0$$

$$\left. \begin{array}{l}
f(E_t, \beta) = \min \left(\beta - E_t, \beta + E^{in}, K \right) \\
g(E_t, \beta) = 0
\end{array} \right\} \text{ si } E_t < 0$$

La relation de récurrence vérifiée par $v(\tau, \zeta, \beta)$ pour tout $\tau < T$ et tout $\zeta > 0$ s'écrit alors :

$$v(\tau, \zeta, \beta) = \max_{0 \leq \delta_\tau \leq 1} (C^g g(\bar{E}_\tau + \delta_\tau \Delta_\tau, \beta) + v(\tau + 1, \zeta - \delta_\tau, f(\bar{E}_\tau + \delta_\tau \Delta_\tau, \beta))),$$

où $v(\tau, 0, \beta) = C^g g(\bar{E}_\tau, \beta) + v(\tau + 1, 0, f(\bar{E}_\tau, \beta))$ et

$$v(T, \zeta, \beta) = C^g g(\bar{E}_T + \min(\zeta, 1) \Delta_T, \beta).$$

D'après la proposition 2 de la section 3.2, il existe une solution du problème de recours telle que $\delta_t \in \{0, 1\}$ pour tout t . Ainsi nous avons :

$$\begin{aligned}
v(\tau, \zeta, \beta) = \max & (C^g g(\bar{E}_\tau, \beta) + v(\tau + 1, \zeta, f(\bar{E}_\tau, \beta)), \\
& C^g g(\bar{E}_\tau + \Delta_\tau, \beta) + v(\tau + 1, \zeta - 1, f(\bar{E}_\tau + \Delta_\tau, \beta))) \quad (8.1)
\end{aligned}$$

Nous allons maintenant étudier $v(\tau, \zeta, \beta)$ en tant que fonction de β . Plus précisément, nous allons montrer que $v(\tau, \zeta, \beta)$ est une fonction linéaire décroissante

de β , sur un intervalle $[0; B]$, puis constante sur $[B; K]$. Nous étudions d'abord deux cas : $\zeta = 0$ et $\zeta = T - \tau + 1$.

Proposition 21. *Si $\zeta = 0$ ou $\zeta = T - \tau + 1$, alors pour $\tau \in \{1, \dots, T\}$ et $\beta \in [0, K]$, nous avons :*

$$v(\tau, \zeta, \beta) = C - \gamma C^g \min(\beta, B),$$

où $B = \frac{v(\tau, \zeta, 0) - v(\tau, \zeta, K)}{\gamma C^g}$ et $C = v(\tau, \zeta, 0)$.

Démonstration. Dans les deux cas, on se ramène à un problème sans incertitudes : si $\zeta = 0$ (respectivement $T - \tau + 1$), alors $\delta_t = 0$ (respectivement 1) pour tout $t = \tau, \dots, T$. Supposons d'abord que $\zeta = 0$; nous montrons maintenant que pour chaque $\tau \in \{1, \dots, T\}$, il existe deux constantes B et C telles que :

$$v(\tau, 0, \beta) = C - \gamma C^g \min(\beta, B), \quad (8.2)$$

Nous vérifierons alors les expressions de B et C . D'après 8.1, nous avons que pour tout $\tau < T$:

$$v(\tau, 0, \beta) = C^g g(\bar{E}_\tau, \beta) + v(\tau + 1, 0, f(\bar{E}_\tau, \beta)).$$

Nous procédons alors par récurrence de T à 1.

— Étape 1 : $\tau = T$, $v(T, 0, \beta) = C^g g(\bar{E}_T, \beta)$.

Si $\bar{E}_T < 0$, alors $v(T, 0, \beta) = 0$ ($B_1 = 0, C_1 = 0$).

Sinon $v(T, 0, \beta) = C^g \bar{E}_T - \gamma C^g \min\left(\beta, E^{out}, \frac{\bar{E}_T}{\gamma}\right)$ ($B'_1 = \min\left(E^{out}, \frac{\bar{E}_T}{\gamma}\right)$, $C'_1 = C^g \bar{E}_T$.)

— Étape 2 : $\tau = \bar{\tau} + 1$.

Supposons qu'il existe $B_2 \geq 0$ et $C_2 \geq 0$ tels que

$$v(\bar{\tau} + 1, 0, \beta) = C_2 - \gamma C^g \min(\beta, B_2).$$

— Étape 3 : $\tau = \bar{\tau}$,

$$v(\bar{\tau}, 0, \beta) = C^g g(\bar{E}_{\bar{\tau}}, \beta) + C_2 - \gamma C^g \min(f(\bar{E}_{\bar{\tau}}, \beta), B_2).$$

— Si $\bar{E}_{\bar{\tau}} \geq 0$, nous avons :

$$\begin{aligned} v(\bar{\tau}, 0, \beta) &= C^g \bar{E}_{\bar{\tau}} - \gamma C^g \min\left(\beta, E^{out}, \frac{\bar{E}_{\bar{\tau}}}{\gamma}\right) \\ &+ C_2 - \gamma C^g \min\left(\max\left(\beta - \frac{\bar{E}_{\bar{\tau}}}{\gamma}, \beta - E^{out}, 0\right), B_2\right) \\ &= (C^g \bar{E}_{\bar{\tau}} + C_2) \\ &- \gamma C^g \left(\min\left(\beta, E^{out}, \frac{\bar{E}_{\bar{\tau}}}{\gamma}\right) + \min\left(\max\left(\beta - \frac{\bar{E}_{\bar{\tau}}}{\gamma}, \beta - E^{out}, 0\right), B_2\right)\right) \end{aligned}$$

En considérant les trois cas $\beta \leq \min\left(E^{out}, \frac{\bar{E}_{\bar{\tau}}}{\gamma}\right)$,
 $\min\left(E^{out}, \frac{\bar{E}_{\bar{\tau}}}{\gamma}\right) < \beta \leq B_2 + \min\left(E^{out}, \frac{\bar{E}_{\bar{\tau}}}{\gamma}\right)$, et $B_2 + \min\left(E^{out}, \frac{\bar{E}_{\bar{\tau}}}{\gamma}\right) < \beta$, on vérifie facilement que :

$$\begin{aligned} v(\bar{\tau}, 0, \beta) &= (C^g \bar{E}_{\bar{\tau}} + C_2) - \gamma C^g \min\left(\beta, B_2 + \min\left(E^{out}, \frac{\bar{E}_{\bar{\tau}}}{\gamma}\right)\right) \\ &= C_3 - \gamma C^g \min(\beta, B_3) \end{aligned}$$

où $B_3 = \min\left(B_2 + \min\left(E^{out}, \frac{\bar{E}_{\bar{\tau}}}{\gamma}\right), K\right)$ (nous pouvons choisir $B_3 \leq K$ puisque $\beta \leq K$), et $C_3 = (C^g \bar{E}_{\bar{\tau}} + C_2)$.

— Si $\bar{E}_{\bar{\tau}} < 0$, alors :

$$\begin{aligned} v(\bar{\tau}, 0, \beta) &= 0 + C_2 - \gamma C^g \min\left(\min(\beta - \bar{E}_{\bar{\tau}}, \beta + E^{in}, K), B_2\right) \\ &= C_2 - \gamma C^g \min(\beta - \bar{E}_{\bar{\tau}}, \beta + E^{in}, K, B_2) \\ &= C_2 - \gamma C^g \min(\beta + \min(-\bar{E}_{\bar{\tau}}, E^{in}), \min(K, B_2)) \\ &= (C_2 - \gamma C^g \min(-\bar{E}_{\bar{\tau}}, E^{in})) - \gamma C^g (\beta, \min(K, B_2) - \min(-\bar{E}_{\bar{\tau}}, E^{in})) \\ &= C'_3 - \gamma C^g \min(\beta, B'_3) \end{aligned}$$

$$\text{où } \begin{cases} B'_3 = \min(K, B_2) - \min(-\bar{E}_{\bar{\tau}}, E^{in}) \text{ et } C'_3 = C_2 - \gamma C^g \min(-\bar{E}_{\bar{\tau}}, E^{in}) \\ \text{si } \min(K, B_2) \geq \min(-\bar{E}_{\bar{\tau}}, E^{in}) \\ B'_3 = 0 \text{ et} \\ C'_3 = C_2 - \gamma C^g \min(-\bar{E}_{\bar{\tau}}, E^{in}) - \gamma C^g (\min(K, B_2) - \min(-\bar{E}_{\bar{\tau}}, E^{in})) \\ \text{si } \min(K, B_2) < \min(-\bar{E}_{\bar{\tau}}, E^{in}) \end{cases}$$

Vérifions maintenant les expressions de B et C . En prenant $\beta = 0$ dans 8.2, on obtient $C = v(\bar{\tau}, 0, 0)$. De plus, $v(\bar{\tau}, 0, B) = v(\bar{\tau}, 0, 0) - \gamma C^g B = v(\bar{\tau}, 0, K)$, et nous obtenons alors

$$B = \frac{v(\bar{\tau}, 0, 0) - v(\bar{\tau}, 0, K)}{\gamma C^g}.$$

La preuve lorsque $\zeta = T - \tau + 1$ est identique. \square

Nous montrons maintenant que la proposition 21 est en fait vraie pour n'importe quelle valeur de (τ, ζ) .

Proposition 22. *Pour tout τ , $1 \leq \tau \leq T$ et tout ζ , $0 \leq \zeta \leq \bar{\delta}$, il existe $B \geq 0$ et $C \geq 0$ tels que*

$$v(\tau, \zeta, \beta) = C - \gamma C^g \min(\beta, B),$$

pour tout β dans $[0, K]$.

Démonstration. Nous le démontrons par récurrence sur (τ, ζ) .

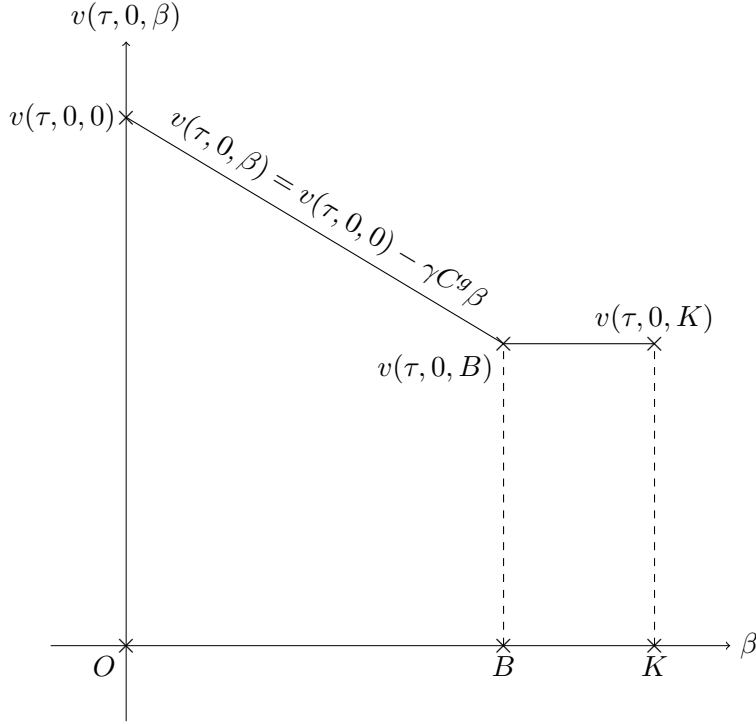


FIGURE 8.1 – Représentation de la fonction $v(\tau, \zeta, B)$, lorsque $\zeta = 0$ ou $\zeta = T - \tau + 1$

- Étape 1 : D'après la proposition 21, la proposition est vraie pour tout τ lorsque $\zeta = 0$ ou $\zeta = T - \tau + 1$ (c.a.d. pour $(\tau, \zeta) \in \{(T, 0), (T - 1, 0), \dots, (1, 0), (T, 1), (T - 1, 2), \dots, (T - i, i + 1), \dots, (T - \bar{\delta} - 1, \bar{\delta})\}$)
- Étape 2 : supposons la proposition vraie pour $(\tau + 1, \bar{\zeta} - 1)$ et pour $(\tau + 1, \bar{\zeta})$, nous avons

$$\begin{aligned} \forall \beta \in [0, K], v(\tau + 1, \bar{\zeta} - 1, \beta) &= C_2 - \gamma C^g \min(\beta, B_2), \\ v(\tau + 1, \bar{\zeta}, \beta) &= C'_2 - \gamma C^g \min(\beta, B'_2), \end{aligned}$$

- Étape 3 : d'après la relation de récurrence 8.1, nous avons :

$$\begin{aligned} v(\tau, \bar{\zeta}, \beta) &= \max(C^g g(\bar{E}_\tau, \beta) + v(\tau + 1, \bar{\zeta}, f(\bar{E}_\tau, \beta)), \\ &\quad C^g g(\bar{E}_\tau + \Delta_\tau, \beta) + v(\tau + 1, \bar{\zeta} - 1, f(\bar{E}_\tau + \Delta_\tau, \beta))). \end{aligned}$$

$$\text{Soit } Q = C^g g(\bar{E}_\tau, \beta) + v(\tau + 1, \bar{\zeta}, f(\bar{E}_\tau, \beta)).$$

— Si $\bar{E}_\tau \geq 0$:

$$\begin{aligned} Q &= C^g \bar{E}_\tau - \gamma C^g \min\left(\beta, E^{out}, \frac{\bar{E}_\tau}{\gamma}\right) \\ &+ C'_2 - \gamma C^g \min\left(\max\left(\beta - \frac{\bar{E}_\tau}{\gamma}, \beta - E^{out}, 0\right), B'_2\right) \\ &= (C^g \bar{E}_\tau + C'_2) \\ &- \gamma C^g \left[\min\left(\beta, E^{out}, \frac{\bar{E}_\tau}{\gamma}\right) + \min\left(\max\left(\beta - \frac{\bar{E}_\tau}{\gamma}, \beta - E^{out}, 0\right), B'_2\right) \right] \end{aligned}$$

Comme dans la preuve de la proposition 21, nous obtenons

$$\begin{aligned} Q &= (C^g \bar{E}_\tau + C'_2) - \gamma C^g \min\left(\beta, B'_2 + \min\left(E^{out}, \frac{\bar{E}_\tau}{\gamma}\right)\right) \\ &= C'_3 - \gamma C^g \min(\beta, B'_3) \end{aligned}$$

où $B'_3 = \min\left(B'_2 + \min\left(E^{out}, \frac{\bar{E}_\tau}{\gamma}\right), K\right)$ et $C'_3 = (C^g \bar{E}_\tau + C'_2)$.

— Si $\bar{E}_\tau < 0$, alors :

$$\begin{aligned} Q &= 0 + C'_2 - \gamma C^g \min\left(\min(\beta - \bar{E}_\tau, \beta + E^{in}, K), B'_2\right) \\ &= C'_2 - \gamma C^g \min(\beta - \bar{E}_\tau, \beta + E^{in}, K, B'_2) \\ &= C_2 - \gamma C^g \min(\beta + \min(-\bar{E}_\tau, E^{in}), \min(K, B'_2)) \\ &= C'_3 - \gamma C^g \min(\beta, B'_3) \end{aligned}$$

où $\begin{cases} B'_3 = \min(K, B'_2) - \min(-\bar{E}_\tau, E^{in}) \text{ et } C'_3 = C'_2 - \gamma C^g \min(-\bar{E}_\tau, E^{in}) \\ \text{si } \min(K, B'_2) \geq \min(-\bar{E}_\tau, E^{in}) \\ B'_3 = 0 \text{ et } C'_3 = C'_2 - \gamma C^g \min(-\bar{E}_\tau, E^{in}) - \gamma C^g (\min(K, B'_2) - \min(-\bar{E}_\tau, E^{in})) \\ \text{si } \min(K, B'_2) < \min(-\bar{E}_\tau, E^{in}) \end{cases}$

Soit $R = C^g g(\bar{E}_\tau + \Delta_\tau, \beta) + v(\tau + 1, \bar{\zeta} - 1, f(\bar{E}_\tau + \Delta_\tau, \beta))$. Nous prouvons de la même manière qu'il existe $C_3'' \geq 0$, et $B_3'' \geq 0$, tels que

$$R = C_3'' - \gamma C^g \min(\beta, B_3'').$$

Ainsi,

$$v(\tau, \bar{\zeta}, \beta) = \max(C'_3 - \gamma C^g \min(\beta, B'_3), C_3'' - \gamma C^g \min(\beta, B_3'')).$$

En étudiant les quatre cas $C'_3 < C''_3$ et $B'_3 < B''_3$; $C'_3 < C''_3$ et $B'_3 \geq B''_3$; $C'_3 \geq C''_3$ et $B'_3 < B''_3$; $C'_3 \geq C''_3$ et $B'_3 \geq B''_3$, on vérifie aisément qu'il existe $B_3, B_3 \in [B'_3, B''_3]$, tel que :

$$v(\tau, \bar{\zeta}, \beta) = C_3 - \gamma C^g \min(\beta, B_3),$$

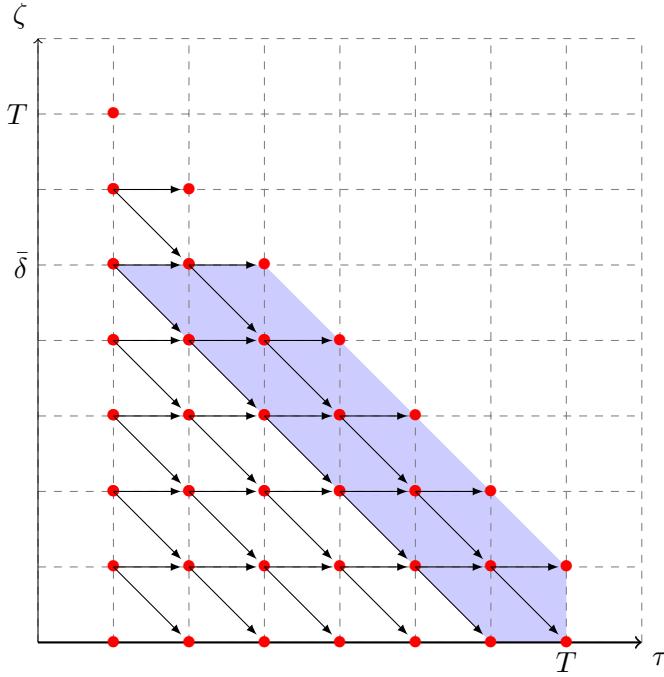


FIGURE 8.2 – Schéma de l'algorithme de programmation dynamique

où $C_3 = \max(C'_3, C''_3)$. Nous terminons cette récurrence en nous référant au schéma de la figure 8.2. En effet, d'après la proposition 21, la proposition est vraie pour $(\tau, \zeta) \in \{(T, 0), (T, 1)\}$, et il est facile de vérifier que si la proposition est vraie pour $(\tau, \zeta) \in \{(\bar{\tau} + 1, 0), (\bar{\tau} + 1, 1), \dots, (\bar{\tau} + 1, T - (\bar{\tau} + 1) + 1)\}$, alors elle l'est aussi pour $(\tau, \zeta) \in \{(\bar{\tau}, 0), (\bar{\tau}, 1), \dots, (\bar{\tau}, T - \bar{\tau})\}$ et pour $(\tau, \zeta) = (\bar{\tau}, T - \bar{\tau} + 1)$, d'après la proposition 21.

□

Remarquons que la proposition 22 nous permet de calculer facilement $v(\tau, \zeta, \beta)$. En effet, comme dans la preuve de la proposition 21, les deux constantes B et C sont égales à :

$$B = \frac{v(\tau, \zeta, 0) - v(\tau, \zeta, K)}{\gamma C^g}$$

$$C = v(\tau, \zeta, 0)$$

Ainsi, chaque fonction $v(\tau, \zeta, \beta)$ de β est entièrement déterminée par les valeurs de $v(\tau, \zeta, 0)$ et de $v(\tau, \zeta, K)$.

Nous pouvons alors utiliser deux algorithmes de programmation dynamique imbriqués pour calculer les valeurs de $v(\tau, \zeta, 0)$ et de $v(\tau, \zeta, K)$ en suivant le schéma indiqué sur la figure 8.2. Remarquons que, le calcul des états initiaux se faisant en $O(T^2)$, la complexité de l'algorithme est en $O(\bar{\delta}(T - \bar{\delta} + 1) + T^2)$. Or, puisque

$\bar{\delta} \leq T$, cet algorithme est bien polynomial.

Pour résoudre le problème robuste, nous devons résoudre le problème de recours linéarisé $LDR(x)$. Or celui-ci peut lui aussi être résolu en temps polynomial. En effet, un fois que les valeurs optimales de δ_t , $t = 1, \dots, T$, sont obtenues par l'algorithme de programmation dynamique, nous n'avons plus qu'à fixer les variables δ_t , de $LDR(x)$, à ces valeurs pour pouvoir résoudre $LDR(x)$ en temps polynomial.

8.2 Comparaison avec l'approche précédente

Nous comparons maintenant la résolution du problème robuste, en résolvant le problème de recours par l'algorithme de programmation dynamique ou par CPLEX. En Table 8.1, nous donnons les temps de calcul CPU, en secondes, des deux approches pour différentes valeurs de $\bar{\delta}$. Nous considérons ici que l'incertitude ne concerne que la demande

	$\bar{\delta}$						
	0	100	500	700	850	1000	8760
CPU(s) programmation dynamique	82	90	97	95	94	102	67
CPU(s) Cplex	46	88	7422	375	204	65	22

TABLE 8.1 – Comparaison des deux approches du point de vue du temps CPU

On remarque que pour des valeurs intermédiaires de $\bar{\delta}$, l'approche par la programmation dynamique est beaucoup plus rapide que celle utilisant Cplex. En particulier, pour certaines valeurs de $\bar{\delta}$ entre 0 et 850, l'approche utilisant Cplex peut prendre plusieurs heures pour aboutir, alors que l'approche par la programmation dynamique ne demande que quelques minutes. De plus, nous remarquons que, pour l'approche utilisant Cplex, le temps CPU augmente d'abord pour $\bar{\delta} \leq 850$, puis diminue. En effet, quand $\bar{\delta} > 850$, de moins en moins de noeuds sont explorés dans le branch and bound.

Chapitre 9

Généralisations et Améliorations

Dans ce chapitre, nous présentons quelques améliorations pour la modélisation et la résolution de notre problème de conception de parc hybride. Dans un premier temps, nous cherchons à enrichir l'ensemble \mathcal{D} , qui modélise l'incertitude sur la demande D , avec de nouvelles contraintes. En effet, jusqu'à présent, les valeurs des demandes à chaque pas de temps étaient supposées indépendantes les unes des autres, ce qui évidemment n'est pas le cas en pratique. Nous construirons ensuite des instances "réalistes" pour un lieu en France, en soignant particulièrement le réalisme des ordres de grandeur. Enfin, nous présenterons quelques améliorations possibles de l'algorithme de génération de contraintes, servant à résoudre le problème robuste, dans le but de traiter des instances de grande taille en un temps raisonnable.

9.1 Ajout de contraintes sur le polyèdre d'incertitude

Les calculs avec les instances de la section 7.4 ont abouti à un constat : en faisant varier le budget d'incertitude, $\bar{\delta}$, pour la demande, la solution optimale est équivalente à la solution déterministe du pire cas à longueur d'année, dès que ce budget atteint environ 10 % du nombre total de périodes. Cela réduit d'autant la marge d'optimisation, puisqu'il n'est pas forcément réaliste de réduire autant $\bar{\delta}$. Il se peut que cela soit dû à un pire cas accolant les périodes de pointe (les périodes t où $\delta_t^* = 1$), aboutissant à un état de décharge chronique des batteries causant un recours accru au générateur, avec les augmentations de coût que cela suppose. L'optimisation consiste alors à augmenter le nombre de batteries, panneaux solaires et éoliennes jusqu'à atteindre un minimum du coût global.

Pour l'énergie solaire et l'énergie éolienne récupérées, ce phénomène suit bien évidemment les phénomènes météorologiques, qui s'étendent sur une échelle de temps caractéristique de quelques jours, liée à la succession des masses d'air séparées par les fronts froids ou chauds.

La demande quant à elle, suit de moins près ces phénomènes météorologiques.

Toutefois, en France, le fort développement du chauffage électrique basique (par convecteur à résistance) a rendu la demande très sensible aux baisses températures (c.f. [2, 3]). Ainsi un gradient thermique d’hiver montre une forte corrélation (en dessous de 15°C) de la température et de la puissance appelée : 2,3 GW en plus par degrés en moins. Cette méthode sera explicitée et argumentée dans la partie suivante.

Par conséquent, les pointes de demande seront alignées dans cette étude sur les phénomènes de température (grands froids mais aussi canicules, qui causent des pointes de climatisation déjà palpables aux Etats-Unis).

La notion de température moyenne utilisée ci-après s’entend comme une moyenne des températures relevées par 11 stations de référence réparties sur la France.



La courbe noire indique la consommation, la courbe rouge la température et les bâtonnets verts l’écart aux températures moyennes saisonnières (moyennes sur 30 ans), ici pour le mois de décembre 2011.

Soit τ et τ_p deux entiers fixés. Nous définissons les contraintes suivantes :

$$\forall t_0 \in \{1, \dots, T - \tau + 1\}, \sum_{t=t_0}^{t_0+\tau-1} \delta_t \leq \tau_p, \quad (9.1)$$

qui interdisent qu’il y ait plus de τ_p périodes de pointe sur une fenêtre de τ périodes. En pratique si l’année est divisée en périodes d’une heure, nous pouvons prendre, en vue des courbes ci-dessus, $\tau = 21 * 24$ et $\tau_p = 15 * 24$.

Cependant, avant d’ajouter les contraintes 9.1 à l’ensemble \mathcal{D} , nous devons vérifier que l’ajout de tels contraintes ne modifie pas le caractère 0 – 1 des scénarios

extrêmes δ_t^* . En effet, afin de pouvoir linéariser le problème de recours, il est primordial de montrer que l'on peut se ramener à une solution optimale, (δ^*, y^*) , du problème de recours où $\delta_t^* \in \{0, 1\}$, $\forall t = 1, \dots, T$.

Soit \mathcal{D} le nouvel ensemble d'incertitude enrichi de ces nouvelles contraintes. Alors les contraintes sur le vecteur δ s'écrivent :

$$\begin{pmatrix} 1 & 0 & \dots & 0 \\ & & \ddots & \\ 0 & 0 & \dots & 1 \\ 1 & \dots & \dots & 1 \\ 1 & \dots & 1 & \\ & \ddots & \dots & \ddots \\ & & 1 & \dots & 1 \end{pmatrix} \delta \leq \begin{pmatrix} 1 \\ \vdots \\ 1 \\ \bar{\delta} \\ \tau_p \\ \vdots \\ \tau_p \end{pmatrix}$$

La matrice ci-dessus étant une matrice d'intervalle, elle est totalement unimodulaire. On peut donc se limiter aux solutions optimales, δ^* , entières, c'est à dire $\delta_t^* \in \{0, 1\}$, $\forall t = 1, \dots, T$.

Nous pouvons donc bien enrichir l'ensemble \mathcal{D} de cette ensemble de contraintes permettant de limiter à τ_p le nombre de périodes de pointe, sur un ensemble de τ périodes consécutives.

9.2 Construction d'instances, choix de données numériques

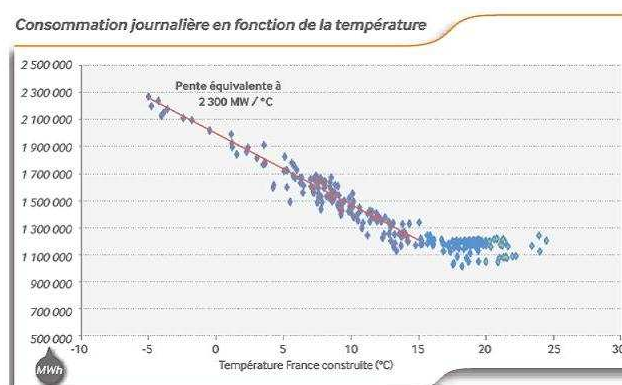
Dans cette section, nous cherchons à construire des instances réalistes pour notre modèle.

La demande a été tirée d'un profil "France", recalé avec l'influence de la température des lieux étudiés, puis mis à une échelle de 10kW, tandis que l'énergie solaire et éolienne ont été tirées de statistiques météorologiques disponibles. Enfin, les composants ont été choisis en recalant les coûts par rapport aux valeurs typiques actuelles.

Suite au choix de la France pour les amplitudes de demande et les pointes d'hiver, des sites restaient à choisir afin d'établir des modèles à l'échelle locale, puisque le parc hybride a pour but d'alimenter des lieux isolés. Nous avons choisi Perpignan, situé à l'extrême sud, et qui bénéficie d'un régime de vent intéressant (moyenne annuelle 5,4 m/s, régime dominant de nord-ouest par la Tramontane), et surtout d'un fort ensoleillement de plus de 2000 heures par an.

9.2.1 Sensibilité à la température

La grande thermosensibilité de la consommation électrique en France génère de fortes amplitudes de puissance appelée entre les creux de nuits d'été (30 GW) et les superpointes d'hiver (102 GW le 8/02/2012 à 19 heures). La demande est même essentiellement sensible à la température, comme le montre ce graphique (c.f. [3]), qui met en évidence la forte corrélation en-dessous de 15°C.



Cela permet de définir un gradient d'hiver. Mais un phénomène analogue bien que de moindre ampleur se développe en été, lorsque les climatisations génèrent un appel de puissance qui croît quasiment linéairement avec la température au dessus de 25°C. C'est même le principal phénomène de surdemande aux États-Unis, où de grands black-out sont survenus durant des vagues de chaleur. Nous retiendrons donc :

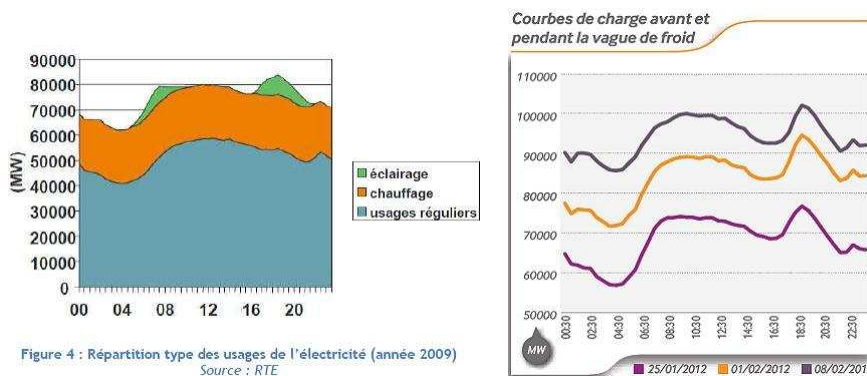
un gradient d'hiver de 2,3 GW appelés en plus pour chaque degré en moins en-dessous de 15°C ;

un gradient d'été de 0,5 GW appelés en plus pour chaque degré en plus au-dessus de 25°C.

9.2.2 Échelles de temps

Cette question sera d'importance pour la suite, afin d'alléger les calculs, par réduction du nombre de variables entières.

Dû à l'extrême variation de la demande, il ne semble pas opportun de discrétiser la période d'étude en des périodes de plus d'une heure. Ainsi, le profil de demande moyenne \bar{D}_t sera établi heure par heure. Par contre la surdemande, qui est essentiellement (exclusivement ici) liée aux écarts de températures extrêmes par rapport à la moyenne agit différemment : elle translate en bloc la courbe moyenne d'une quantité quasiment constante à l'échelle d'un jour. On parle d'un "ruban de chauffage", illustré par les deux graphiques ci-dessous. Effectivement, les bâtiments jouissent d'une inertie thermique qui lisse pratiquement la consommation de chauffage à la température moyenne journalière.



Le profil d'écart de demande extrême Δ_t est donc établi jour par jour en fonction des écarts extrêmes de températures par rapport à la moyenne.

Ainsi, le procédé, pour calculer les demandes moyennes et les écarts de demandes extrêmes, est le suivant :

1. Récupération des données RTE de puissance appelée.
2. Construction d'une demande théorique neutre en température D^0 en retranchant les gradients thermiques.
3. Récupération de données météorologiques : températures moyennes, journalières et extrêmes sur les 30 dernières années (à Perpignan)
4. Construction de la demande moyenne en additionnant la part thermosensible du lieu à la demande neutre.
5. Construction de l'écart extrême en utilisant de même les températures extrêmes.
6. Mise à l'échelle par application générale d'un facteur 10^{-4} .

9.2.3 Modèles de ressource solaire et éolienne

La prise en compte de la ressource solaire et éolienne s'inspire d'une étude de faisabilité de centrales hybrides isolées à St John (Terre-Neuve, Canada)[25], qui a utilisé le logiciel HOMER Energy (c.f. [4]) pour générer des séries pseudo-aléatoires de puissance de radiation et de vitesse de vent, à partir de moyennes mensuelles accessibles sur internet. Ceci détermine alors un profil moyen.

Ressource solaire

Le flux de radiation solaire, maximal, reçu sur un plan perpendiculaire à la direction du soleil est de $1 \text{ kW}/\text{m}^2$. Cette valeur est ensuite affectée d'un coefficient lié au sinus de l'angle d'incidence réel (élévation du soleil selon l'heure et la saison, azimut selon l'heure, inclinaison et orientation du panneau) et à la nébulosité (indice de clarté). Puis, le résultat est multiplié par un facteur lié au rendement de ces panneaux, puis un facteur lié au rendement de l'onduleur.

Les valeurs moyennes mensuelles des flux de radiations sont exprimées en kWh/m²/J, soit une énergie reçue par 24 heures sur une unité de surface. La feuille de calcul disponible sur le site de l'Institut National d'Energie Solaire (INES) permet d'obtenir ces données pour diverses stations météorologiques, dont Perpignan. L'angle d'inclinaison des panneaux a été choisi par simulations successives de telle sorte que le flux moyen annuel soit proche de son maximum, tout en privilégiant le plus possible les mois d'hiver où la demande est élevée. C'est pourquoi les inclinaisons retenues ne sont pas directement égales aux latitudes de Perpignan.

Énergie surfacique reçue par jour (moyenne mensuelle) kWh/m ² /J											
J	F	M	A	M	J	J	A	S	O	N	D
Perpignan : latitude 42,7°N, inclinaison 55°											
3.87	4.22	5.04	5.24	5.1	5.14	5.22	4.93	4.79	4.18	3.4	3.06

Ensuite, le logiciel HOMER génère une série pseudo-aléatoire de puissance surfacique par pas d'une heure à partir de ces moyennes et de la latitude du lieu, afin de rendre compte d'une variabilité d'un jour à l'autre pour cause de nébulosité. Il ne reste plus qu'à multiplier ces valeurs par la surface du panneau, le rendement, le rendement de l'onduleur, pour obtenir \bar{E}_t^s . L'écart défavorable a été fixé à 60% de la valeur moyenne.

Ressource éolienne

Le vent est soumis à une très forte variabilité sur une courte échelle de temps, à tel point que les relevés anémométriques d'étude des régimes sont effectués toutes les dix minutes.

La répartition probabiliste des fréquences d'apparition des vitesses de vent est communément modélisée par la densité, f , de probabilité de Weibull :

$$f(v) = \frac{kv^{k-1}}{c^k} e^{-\left(\frac{v}{c}\right)^k},$$

où k est le paramètre dit de forme et $c = \frac{\text{vitesse moyenne}}{0.89}$ est le paramètre dit d'échelle.

Le paramètre k est habituellement compris entre 1 (forte variabilité) et 3 (vitesse variant peu), et environ 2 dans nos climats. Pour $k = 2$, il s'agit précisément de la distribution de Rayleigh.

En utilisant [1] et [17], nous calculons les paramètres et moyennes mensuelles :

Vitesse moyenne (m/s)											
J	F	M	A	M	J	J	A	S	O	N	D
Perpignan : $k = 2.0$ $c = 6.07$ m/s											
5.5	6.0	6.0	6.0	5.5	5.0	5.5	5.0	5.0	5.0	5.0	5.0

Ensuite le logiciel HOMER génère une série pseudo-aléatoire de vitesse de vent par pas d'une heure à partir de ces moyennes et de la latitude du lieu, afin de rendre compte d'une variabilité d'une heure à l'autre. Enfin il ne reste qu'à convertir ces vitesses en puissance utile, au travers de la courbe caractéristique de l'éolienne.

L'écart défavorable a été retenu à 80% de la valeur moyenne.

9.2.4 Choix des composants et calcul des coûts

Panneaux solaire

Le modèle retenu est celui proposé sur le site EdF-EnR, qui propose diverses combinaisons à partir de modules de 250 Wc, soit $2,1 m^2$ avec un rendement de 12%. Le rendement de l'onduleur a été fixé à 90%. Le coût d'installation typique est d'environ 4000 euros par kWc. En fixant une durée de vie estimée à vingt ans, il vient un coût fixe annuel d'investissement / renouvellement $C^s = 50.4$ euros. Le nombre maximal de panneaux a été fixé à 200.

Eoliennes

Deux éoliennes dont les courbes de puissance sont connues (c.f. [4]) seront étudiées : les modèles Excel-R (7,5 kW) et Excel-S (10 kW) de Bergey Windpower Co.

En ajoutant des frais de maintenance annuels de 1,5% de l'investissement initial et en considérant une durée de vie de vingt ans, les coûts fixes annuels retenus sont : $C^w = 2665$ euros par an pour le modèle 7,5 kW ; $C^w = 2925$ euros par an pour le modèle 10 kW. Le nombre maximal d'éoliennes a été fixé à 15.

Batteries

Les modèles étudiés sont des batteries de grande capacité à longue durée de vie, conçues pour cette utilisation de stockage. Nous considérerons la Rolls-Surette 6CS25P (6V) et la 4KS25P (4V) dont les caractéristiques techniques sont disponibles dans le logiciel HOMER.

Générateur auxiliaire

Dans le cadre de cette étude, le coût fixe d'installation du groupe électrogène n'est pas pris en compte, car il est considéré égal sur toutes les instances, en dimensionnant la puissance installée proche de la plus forte demande.

Le coût d'usage est déduit d'une fourchette de consommation typique de 0,28 à 0,4 litre de gazole par kWh produit. La borne supérieure sera retenue, pour prendre en compte un usage ponctuel, typique du recours.

Considérant un prix du gazole de 1,4 euros le litre, et un coût de maintenance estimé à 5 centimes d'euro par kWh, l'on retiendra $C^g = 0.6$ euros par kWh.

9.3 Résultats numériques

Dans cette section, nous testons notre approche sur les instances qui ont été définies en 9.2.

Les tests sont divisés en deux parties : dans un premier temps, nous testerons la pertinence d'agréger les variables δ_t par 24 heures ; puis nous testerons l'impact de la contrainte dite "glissante" (c.f. 9.1), ajoutée en 9.1, en fonction du budget de l'incertitude $\bar{\delta}$.

Comme en 7.4, les tests ont été effectués sur un Intel Nehalem XEON 5570 à 2.93 GHz ayant 24 Go de RAM.

9.3.1 Pertinence d'agréger les variables entières δ_t par 24 heures

En agrégeant les δ_t par blocs d'un jour, nous pouvons espérer diminuer de façon conséquente le temps de calcul d'une solution robuste. Les calculs suivants ont pour but d'estimer l'erreur afin de vérifier si elle est acceptable. Ce procédé n'est pas sans fondement, puisqu'il a déjà été constaté que les surdemandes, ou les creux d'énergie solaire ou éolienne, correspondaient à des phénomènes à l'échelle de quelques jours.

Nous donnons en Table 9.1 l'écart entre les solutions optimales lorsque les variables δ_t sont agrégées par 24 heures ou non. Ces tests sont réalisés avec l'instance "Perpignan" pour plusieurs valeurs de Δ . Par conséquent, il est possible d'agréger

Test	δ_t par heure	δ_t par jour	Ecart coût
$\Delta_t = 10\% \bar{D}_t$ Temps CPU(s)	$x^s = 15, x^w = 2, x^b = 11$ Coût=19837,8 772	$x^s = 15, x^w = 2, x^b = 10$ Coût=19768,6 365	0,34%
$\Delta_t = 30\% \bar{D}_t$ Temps CPU(s)	$x^s = 15, x^w = 2, x^b = 10$ Coût=21611,5 574	$x^s = 15, x^w = 2, x^b = 9$ Coût=21426 340	0,866%
Δ_t défini à partir des températures Temps CPU(s)	$x^s = 15, x^w = 2, x^b = 11$ Coût=21565,7 365	$x^s = 15, x^w = 2, x^b = 9$ Coût=21374,7 154	0,891%

TABLE 9.1 – Ecart entre les solutions optimales lorsque les δ_t sont agrégés ou non

les variables δ_t par 24 heures, au prix d'une sous-estimation du coût inférieure à 1%. En outre, ce modèle par jour correspond mieux au problème physique qu'un

comportement erratique d'heure en heure : agréger ces variables revient à imposer des contraintes qui rendent le modèle plus réaliste.

9.3.2 Impact de la contrainte glissante

Nous étudions maintenant quelle est l'impact de la contrainte dite "glissante", définie en 9.1, sur la valeur de la solution optimale robuste. Les tests ont été effectués avec les paramètres suivants :

- Les données Perpignan avec Δ_t défini selon les températures, $\Phi_t = 60\% \bar{E}_t^s$, $\Omega_t = 80\% \bar{E}_t^w$, où Φ et Ω sont définis en section 7.3 ;
- $\bar{\delta}$ ou $\bar{\phi}$ ou $\bar{\omega}$ appartenant à $\{0, 50, 100, 150, 200, 250, 300, 365\}$;
- sans contrainte glissante, 9.1, ou avec (pour les budgets de 50 à 250).

Test	Sans 9.1	Avec 9.1	Impact de 9.1 %
$\bar{\delta} = 0$	$x^s = 95, x^w = 1, x^b = 28$ 16173	Sans objet	
$\bar{\delta} = 50$	$x^s = 96, x^w = 1, x^b = 27$ Coût=17518,3	$x^s = 98, x^w = 1, x^b = 27$ Coût=17518,3	0%
$\bar{\delta} = 100$	$x^s = 100, x^w = 1, x^b = 27$ Coût=18517,4	$x^s = 100, x^w = 1, x^b = 27$ Coût=18476,0	0, 224%
$\bar{\delta} = 150$	$x^s = 106, x^w = 1, x^b = 28$ Coût=19244,1	$x^s = 107, x^w = 1, x^b = 28$ Coût=18943,2	1, 588%
$\bar{\delta} = 200$	$x^s = 111, x^w = 1, x^b = 28$ Coût=19633	$x^s = 106, x^w = 1, x^b = 29$ Coût=19072	2, 941%
$\bar{\delta} = 250$	$x^s = 113, x^w = 1, x^b = 30$ Coût=19794	$x^s = 109, x^w = 1, x^b = 30$ Coût=19103	3, 617%
$\bar{\delta} = 300$	$x^s = 116, x^w = 1, x^b = 31$ Coût=19836,1	Sans objet	
$\bar{\delta} = 365$	$x^s = 116, x^w = 1, x^b = 31$ Coût=19836,1	Sans objet	

TABLE 9.2 – Impact de la contrainte glissante lorsque l'incertitude porte sur la demande

Nous en déduisons que la contrainte glissante en "15 jours sur 21" n'apporte aucune amélioration notable pour des budgets d'incertitude inférieures à 200. Cette contrainte peut donc être négligée.

De plus, la valeur du coût robuste en fonction de $\bar{\delta}$, dépend bien plus de la valeur des données de la demande et des énergies renouvelables que de la présence

Test	Sans 9.1	Avec 9.1	Impact de 9.1 %
$\bar{\phi} = 0$	$x^s = 98, x^w = 1, x^b = 28$ 16173,4	Sans objet	
$\bar{\phi} = 50$	$x^s = 102, x^w = 1, x^b = 23$ Coût=17494,0	$x^s = 83, x^w = 1, x^b = 24$ Coût=17484,8	0,053%
$\bar{\phi} = 100$	$x^s = 61, x^w = 2, x^b = 15$ Coût=18535,3	$x^s = 86, x^w = 1, x^b = 21$ Coût=18523,3	0,065%
$\bar{\phi} = 150$	$x^s = 57, x^w = 2, x^b = 16$ Coût=19190	$x^s = 107, x^w = 1, x^b = 23$ Coût=19182,5	0,039%
$\bar{\phi} = 200$	$x^s = 43, x^w = 2, x^b = 16$ Coût=19635,6	$x^s = 56, x^w = 2, x^b = 14$ Coût=19601,3	0,175%
$\bar{\phi} = 250$	$x^s = 53, x^w = 2, x^b = 13$ Coût=19905,9	$x^s = 53, x^w = 2, x^b = 14$ Coût=19808,2	0,493%
$\bar{\phi} = 300$	$x^s = 53, x^w = 1, x^b = 11$ Coût=20027,4	Sans objet	
$\bar{\phi} = 365$	$x^s = 116, x^w = 1, x^b = 31$ Coût=20042,6	Sans objet	

TABLE 9.3 – Impact de la contrainte glissante lorsque l'incertitude porte sur l'énergie solaire

de la contrainte glissante : ainsi en section 7.4, nous avons vu que le seuil à partir duquel le coût de la solution robuste restait constant était d'environ $\bar{\delta} = 10\%T$, or nous voyons ici que pour l'instance Perpignan, ce seuil est supérieure à $\bar{\delta} = 65\%T$.

Test	Sans 9.1	Avec 9.1	Impact de 9.1 %
$\bar{\omega} = 0$	$x^s = 95, x^w = 1, x^b = 28$ 16173,4	Sans objet	
$\bar{\omega} = 50$	$x^s = 134, x^w = 0, x^b = 35$ Coût=17196,1	$x^s = 134, x^w = 0, x^b = 35$ Coût=17196,1	0%
$\bar{\omega} = 100$	$x^s = 136, x^w = 0, x^b = 36$ Coût=17196,1	$x^s = 136, x^w = 0, x^b = 36$ Coût=17196,1	0%
$\bar{\omega} = 100$	$x^s = 136, x^w = 0, x^b = 36$ Coût=17196,1	$x^s = 136, x^w = 0, x^b = 36$ Coût=17196,1	0%
$\bar{\omega} = 100$	$x^s = 136, x^w = 0, x^b = 36$ Coût=17196,1	$x^s = 136, x^w = 0, x^b = 36$ Coût=17196,1	0%
$\bar{\omega} = 100$	$x^s = 136, x^w = 0, x^b = 36$ Coût=17196,1	$x^s = 136, x^w = 0, x^b = 36$ Coût=17196,1	0%
$\bar{\omega} = 100$	$x^s = 136, x^w = 0, x^b = 36$ Coût=17196,1	$x^s = 136, x^w = 0, x^b = 36$ Coût=17196,1	0%
$\bar{\omega} = 100$	$x^s = 136, x^w = 0, x^b = 36$ Coût=17196,1	$x^s = 136, x^w = 0, x^b = 36$ Coût=17196,1	0%

TABLE 9.4 – Impact de la contrainte glissante lorsque l’incertitude porte sur l’énergie éolienne

Chapitre 10

Conclusion

Dans cette thèse, nous avons étudié les programmes linéaires mixtes robustes bi-niveaux avec des variables de recours continues et la conception d'un parc hybride de production électrique. Dans le chapitre 3, nous nous sommes limités à un problème qui vérifie la propriété de recours complet. Après avoir montré que l'on pouvait se ramener à un programme robuste borné, nous avons rappelé, lorsque l'incertitude ne concerne que le membre droit des contraintes, comment résoudre le problème robuste à l'aide d'un algorithme de génération de contraintes. Cependant, en pratique, de nombreux problèmes ne vérifient pas cette hypothèse. Ainsi dans le chapitre suivant, nous nous avons montré qu'il était toujours possible, en ajoutant de nouvelles variables de recours, de construire un programme robuste qui vérifie la propriété de recours complet et qui est équivalent au programme robuste initial. Pour cela, il est parfois nécessaire d'introduire de très grands coefficients dans les modèles mathématiques, qui peuvent alors devenir difficiles à manier. Nous avons cependant pu tester l'efficacité de notre approche sur un problème de production ne vérifiant pas l'hypothèse de recours complet ; nous avons pu résoudre rapidement des instances qui comportent plus de cinq cents variables et contraintes. Dans le chapitre 5, nous nous sommes placés dans le cas où l'ensemble d'incertitude \mathcal{D} est un polytope quelconque. En effet, l'approche développée dans les chapitres précédents ne permet pas de résoudre les cas où les points extrêmes de \mathcal{D} n'appartiennent pas à l'ensemble $\{0, 1\}$. Nous avons d'abord montré, en examinant les sommets de \mathcal{D} , que le problème de recours pouvait être résolu en temps polynomial lorsque le nombre de contraintes était fixé. Nous en avons ensuite déduit que le problème robuste pouvait se résoudre en temps polynomial quand un autre paramètre est aussi fixé. Cependant cette approche théorique ne permet pas de résoudre des problèmes de grandes tailles. Il serait donc intéressant d'étudier une approche plus efficace pour traiter le cas général. Dans le chapitre 6, nous avons cherché à vérifier la pertinence de l'ensemble d'incertitude \mathcal{D} . Nous avons d'abord borné supérieurement la probabilité qu'un scénario d sorte de \mathcal{D} . Nous avons alors remarqué que lorsque le paramètre d'incertitude $\bar{\delta}$ était légèrement supérieur au nombre de contraintes divisé par deux, cette probabilité devenait très faible. Nous avons en-

suite montré comment calculer la probabilité qu'une solution du problème robuste dépasse un certain coût, notre approche se basant sur le calcul du volume d'un polytope, que l'on approxime par une union d'hypercubes. Cependant cette approche n'a pas pu être testée, et il faudrait comparer notre algorithme de calcul de volume avec d'autres méthodes afin d'étudier celle qui convient le mieux à notre problème. Remarquons que les résultats qui ont été présentés dans cette partie ne s'étendent pas au cas où les variables de recours sont entières. Il serait donc intéressant de rechercher d'autres méthodes permettant de résoudre les cas où les variables de recours comportent une partie entière.

Dans la deuxième partie, nous nous sommes intéressés au dimensionnement d'un parc de production électrique constitué d'éoliennes, de panneaux photovoltaïques, de batteries et d'un générateur à diesel. Dans le chapitre 7, nous avons d'abord montré que le problème déterministe pouvait être modélisé par un programme linéaire mixte. Nous avons montré que ce programme appartenait à la classe des problèmes NP-difficiles, lorsque le nombre de variables entières (c'est à dire le nombre de sources d'énergie hybride) du programme n'était pas fixé. Cependant, nous nous sommes aperçus que lorsque les variables entières étaient fixées, les valeurs optimales des variables liées au fonctionnement du parc pouvaient se calculer en temps linéaire. Pour prendre en compte l'incertitude sur la demande et sur la production d'énergie hybride et éolienne, nous avons choisi l'approche robuste développée dans la première partie. Nous avons ensuite testé notre approche sur des instances réelles et avons vérifié que nous pouvions les résoudre en des temps de calcul raisonnables. Nous avons cependant remarqué que la valeur de la solution robuste convergait très rapidement vers son maximum lorsque le paramètre d'incertitude $\bar{\delta}$ augmentait. Dans le chapitre 8, nous avons montré que nous pouvions résoudre le problème de recours associé en un temps polynomial, à l'aide d'un algorithme de programmation dynamique. Nous avons constaté que lorsque seule la demande était incertaine, l'algorithme de programmation dynamique permettait de résoudre le problème de recours, et donc le problème robuste, beaucoup plus rapidement qu'avec l'approche du chapitre précédent. Cependant, l'algorithme utilisant la programmation dynamique ne permet pas de résoudre des instances réelles lorsque l'incertitude touche aussi la production d'énergie éolienne et solaire. En outre, cet algorithme fonctionne uniquement lorsque le coût du générateur auxiliaire reste constant au cours du temps. Il serait donc intéressant d'essayer d'adapter cet algorithme à un problème plus général. Enfin, dans le chapitre 9, nous avons ajouté de nouvelles contraintes à l'ensemble d'incertitude \mathcal{D} , pour mieux modéliser la demande. Cependant nous avons constaté que leur ajout n'avait pas de réelle influence sur le coût de la solution robuste et que nous pouvions, par conséquent, les négliger. Nous avons cependant montré qu'il était possible d'agrèger les variables δ_t par périodes de vingt-quatre heures sans réellement changer le coût de la solution robuste. Pour conclure, remarquons qu'il serait intéressant d'étouffer la modélisation du problème de dimensionnement afin de rendre compte d'un meilleur réalisme : il faudrait prendre en compte les phénomènes non-linéaires qui

ont lieu au niveau de la charge et de la décharge d'énergie dans les batteries ; en outre, nous avons complètement négligé le fait que lorsque l'on place une éolienne, celle-ci peut "faire de l'ombre" à une autre et donc baisser de manière significative sa production d'énergie.

Bibliographie

- [1] <http://fr.windfinder.com/windstats/>. vitesse moyennes mensuelles pour des stations situées en bord de mer.
- [2] La pointe d'électricité en france ... zéro pointé ! dossier de presse, décembre 2009. analyse de la part de chauffage dans la consommation et de la thermosensibilité.
- [3] Prospective et transparence direction economie : Bilan électrique 2012. rapport technique, rte, 1 terrasse bellini - tsa 41000 - 92919 la defence cedex, 2013. partie 1 : analyse des contrastes de la consommation entre été et hiver.
- [4] Homer energy, energy modeling software for hybrid renewable energy systems. available on line : <http://www.homerenergy.com>, 2011.
- [5] B. Ai, H. Yang, H. Shen, and X. Liao. Computer-aided design of pv/wind hybrid system. *Renewable Energy*, 28 :1491–1512, 2003.
- [6] T. Assavapokee, M. Realf, J. Ammons, and I. Hong. Scenario relaxation algorithm for finite scenario based min-max regret and min-max relative regret robust optimization. *Computer and operation research*, 35 :154–163, 2008.
- [7] F. Babonneau, O. Klopfenstein, A. Ouorou, and J.-P. Vial. Robust capacity expansion solutions for telecommunication networks with uncertain demands. http://www.optimization-online.org/DB_HTML/2010/08/2712.html, 2010.
- [8] C. Bandi and D. Bertsimas. Tractable stochastic analysis in high dimensions via robust optimization. *Mathematical Programming*, 134 :23–70, 2012.
- [9] M. Belhamel, S. Moussa, and A. Kaabeche. Production d'électricité au moyen d'un système hybride (éolien-photovoltaïque-diesel). *Renewable and Sustainable Energy Reviews*, pages 49–54, 2002.
- [10] A. Ben-Tal, El Ghaoui, and A. Nemirovski. *Robust Optimization*. Princeton Series in Applied Mathematics, 2009.
- [11] A. Ben-Tal and A. Nemirovski. Robust solutions of uncertain linear programs. *Operations Research Letters*, 25(1) :1–13, 1999.
- [12] G. Bennett. Probability inequalities for the sum of independent random variables. *Journal of the American Statistical Association*, 57(297) :33–45, 1962.

- [13] D. Bertsimas and M. Sim. The price of robustness. *Operations Research*, 52 :35–53, 2004.
- [14] B. Büeler, A. Enge, and K. Fukuda. Exact volume computation for polytope : A practical study. *Polytopes - Combinatorics and Computation, DMV Seminar*, 29 :131–154, 2000.
- [15] S. Diaf, D. Diaf, M. Belhamel, M. Haddadi, and A. Louche. A methodology for optimal sizing of autonomous hybrid pv/wind system. *Energy Policy*, 35-11 :5708–5718, 2007.
- [16] L. Ferrer-Marti, B. Domenech, A. Garcia-Villoria, and R. Pastor. A milp model to design hybrid wind-photovoltaic isolated rural electrification projects in developing countries. *European Journal of Operational Research*, 226(2) :293–300, 2013.
- [17] Paul François. Le vent : un aspect de la spécificité climatique de la france du nord. *Revue de géographie de Lyon*, 65(2) :80 – 84, 1990.
- [18] V. Gabrel, M. Lacroix, C. Murat, and N. Remli. Robust location transportation problems under uncertain demands. *Discrete Applied Mathematics*, available online.
- [19] M. R. Garey and D. S. Johnson. *Computers and Intractability : A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, 1979.
- [20] B. Grünbaum. *Convex Polytopes*. Springer-Verlag, 2nd edition, 2003.
- [21] A. Gupta, R. P. Saini, and M. P. Sharma. Modelling of hybrid energy system. *Renewable Energy*, 36 :459–481, 2011.
- [22] A. Hertz, O. Marcotte, A. Mdimagh, M. Carreau, and F. Welt. Optimizing the design of a wind farm collection network. *INFOR*, 50(2) :95–104, 2012.
- [23] JR H.W. Lenstra. Integer programming with a fixed number of variables. *Mathematics of Operations Research*, 8 :538–547, 11 1983.
- [24] F. K. Hwang, S. Onn, and U. G. Rothblum. A polynomial time algorithm for shaped partition problems. *SIAM J. Optim*, 10 :70–81, 1999.
- [25] M. J. Khan and M. T. Iqbal. Pre-feasibility study of stand-alone hybrid energy systems for applications in newfoundland. *Renewable energy*, 30 :835–854, 2005.
- [26] Bernhard Korte and Jens Vygen. *Combinatorial Optimization : Theory and Algorithms*. Springer, 2007.
- [27] P. Kouvelis and G. Yu. *Robust Discrete Optimization and its Applications*. Kluwer Academic Publishers, 1997.
- [28] T.W. Lambert and D.C. Hittle. Optimization of autonomous village electrification systems by simulated annealing. *Solar Energy*, 68 :121–132, 2000.
- [29] M. Minoux. On 2-stage robust LP with RHS uncertainty : complexity results and applications. *Journal of Global Optimization*, 49 :521–537, 2011.

- [30] H. Morais, P. Kadar, P. Faria, Z.A. Vale, and H.M. Khodr. Optimal scheduling of a renewable micro-grid in an isolated load area using mixed-integer linear programming. *Renewable Energy*, 35 :151–156, 2010.
- [31] I. Mustakerov and D. Borissova. Wind turbines type and number choice using combinatorial optimization, 2010.
- [32] Shmuel Onn. *Nonlinear Discrete Optimization An Algorithmic Theory*. European Mathematical Society, 2010.
- [33] A. Rajendra Prasad and E. Natarajan. Optimization of integrated photovoltaic-wind power generation systems with battery storage. *Energy*, 31 :1943–1954, 2006.
- [34] A. L. Soyster. Convex programming with set-inclusive constraints and applications to inexact linear programming. *Operations Research*, 21(5) :1154–1157, 1973.
- [35] A. Thiele, T. Terry, and M. Epelman. Robust linear optimization with recourse. http://www.optimization-online.org/DB_HTML/2009/03/2263.html, 2010.
- [36] B. Zeng and L. Zhao. Solving two-stage robust optimization problems using a column-and-constraint generation method. *Operations Research Letters*, 41 :457–41, 2013.

Résumé

Dans cette thèse, nous nous intéressons à l'optimisation robuste. Plus précisément, nous nous intéresserons aux problèmes linéaires mixtes bi-niveaux, c'est à dire aux problèmes dans lesquels le processus de décision est divisé en deux parties : dans un premier temps, les valeurs optimales des variables dites "de décisions" seront calculées ; puis, une fois que l'incertitude sur les données est levée, nous calculerons les valeurs des variables dites "de recours". Dans cette thèse, nous nous limiterons au cas où les variables de deuxième étape, dites "de recours", sont continues.

Dans la première partie de cette thèse, nous nous concentrerons sur l'étude théorique de tels problèmes. Nous commencerons par résoudre un problème linéaire simplifié dans lequel l'incertitude porte seulement sur le membre droit des contraintes, et est modélisée par un polytope bien particulier. Nous supposons en outre que le problème vérifie une propriété dite "de recours complet", qui assure que, quelles que soient les valeurs prises par les variables de décisions, si ces dernières sont admissibles, alors le problème admet toujours une solution réalisable, et ce, quelles que soient les valeurs prises par les paramètres incertains. Nous verrons alors une méthode permettant, à partir d'un programme robuste quelconque, de se ramener à un programme robuste équivalent dont le problème déterministe associé vérifie la propriété de recours complet. Avant de traiter le cas général, nous nous limiterons d'abord au cas où les variables de décisions sont entières. Nous testerons alors notre approche sur un problème de production. Ensuite, après avoir remarqué que l'approche développée dans les chapitres précédents ne se généralisait pas naturellement aux polytopes qui n'ont pas des points extrêmes 0-1, nous montrerons comment, en utilisant des propriétés de convexité du problème, résoudre le problème robuste dans le cas général. Nous en déduirons alors des résultats de complexité sur le problème de deuxième étape, et sur le problème robuste. Dans la suite de cette partie nous tenterons d'utiliser au mieux les informations probabilistes que l'on a sur les données aléatoires pour estimer la pertinence de notre ensemble d'incertitude.

Dans la deuxième partie de cette thèse, nous étudierons un problème de conception de parc hybride de production d'électricité. Plus précisément, nous chercherons à optimiser un parc de production électrique constitué d'éoliennes, de panneaux solaires, de batteries et d'un générateur à diesel, destiné à répondre à une demande locale d'énergie électrique. Il s'agit de déterminer le nombre d'éoliennes, de panneaux solaires et de batteries à installer afin de répondre à la demande pour un coût minimum. Cependant, les données du problème sont très aléatoires. En effet, l'énergie produite par une éolienne dépend de la force et de la direction du vent ; celle produite par un panneau solaire, de l'ensoleillement et la demande en électricité peut être liée à la température ou à d'autres paramètres extérieurs. Pour résoudre ce problème, nous commencerons par modéliser le problème déterministe en un programme linéaire mixte. Puis nous appliquerons directement l'approche de la première partie pour résoudre le problème robuste associé. Nous montrerons ensuite que le problème de deuxième étape associé, peut se résoudre en temps polynomial en utilisant un algorithme de programmation dynamique. Enfin, nous donnerons quelques généralisations et améliorations pour notre problème.

mots-clés : Programmation linéaire, Optimisation robuste, Programmation dynamique, Énergie renouvelable

Abstract

Robust optimization is a recent approach to study problems with uncertain data that does not rely on a prerequisite precise probability model but on mild assumptions on the uncertainties involved in the problem. We studied a linear two-stage robust problem with mixed-integer first-stage variables and continuous second stage variables. We considered column wise uncertainty and focused on the case when the problem doesn't satisfy a "full recourse property" which cannot be always satisfied for real problems. We also studied the complexity of the robust problem which is NP-hard and proved that it is actually polynomial solvable when a parameter of the problem is fixed. We then applied this approach to study a stand-alone hybrid system composed of wind turbines, solar photovoltaic panels and batteries. The aim was to determine the optimal number of photovoltaic panels, wind turbines and batteries in order to serve a given demand while minimizing the total cost of investment and use. We also studied some properties of the second stage problem, in particular that the second stage problem can be solvable in polynomial time using dynamic programming.

key-words : Linear Programming, Robust Optimization, Dynamic Programming, Renewable Energy