

École Doctorale Sciences Pour l'Ingénieur Lille 1 Nord-de-France
Laboratoire CRISTAL - Equipe DOLPHIN

Doctorat

THÈSE

pour obtenir le grade de docteur délivré par

Université de Lille 1
Spécialité doctorale "Informatique"

présentée et soutenue publiquement par

Sylvain DUFOURNY

le 13 octobre 2017

**Optimisation de décisions économiques concurrentielles
dans un simulateur de gestion d'entreprise**

Directeur de thèse : **Clarisse DHAENENS**

Jury

Mme Sylvie Mira Bonnardel,	MCF HDR (section 06), École Centrale Lyon	Rapporteur
M. Ameer Soukhal,	Professeur (section 27), Université de Tours	Rapporteur
Mme Laurence DUCHIEN,	Professeur (section 27), Université de Lille	Examineur
M. Ammar OULAMARA,	Professeur (section 27), Université de Lorraine	Examineur

Université de Lille 1
Centre de Recherche en Informatique, Signal et Automatique de Lille
UMR CNRS 9189, F-59655 Villeneuve d'Ascq Cedex, France



Remerciements

Je tiens tout d'abord à remercier Clarisse, ma directrice de thèse, avec qui j'ai eu la chance et le plaisir de travailler pendant ces quelques années. Je la remercie en particulier pour ses qualités humaines, son humilité et ses remarques toujours constructives et pertinentes. Ses nombreux conseils et relectures m'ont permis de finaliser ce projet.

Je souhaite remercier les membres du jury pour l'intérêt qu'ils ont porté à mon travail et les observations qu'ils y ont apportées : Mme MIRA BONNARDEL et M. SOUKHAL, rapporteurs, M. OULAMARA, examinateur, et Mme Laurence DUCHIEN, présidente du jury.

Je remercie mon épouse, Valérie, pour ses relectures rapides et efficaces et pour m'avoir soutenu, encouragé et souvent supporté depuis les débuts de Strat&Logic. Elle était encore fidèle au poste pour cette nouvelle aventure estudiantine. Une grande pensée pour Ninon, Iris et Sarah, mes filles, qui donnent un sens à mon existence.

Je remercie également mes parents pour m'avoir donné le goût des sciences et rendu curieux. Je remercie ma sœur, Laurence, qui, pour la recherche, représente un réel exemple de motivation, de dévouement et de réussite.

Je tiens à remercier Laurent et Jean-Michel, associés de Strat&Logic, présents depuis les débuts, pour m'avoir encouragé dans cette démarche et soutenu pendant sa réalisation.

Je remercie également tous mes amis pour leur soutien. Je ne les citerai pas mais ils se reconnaîtront.



Table des matières

Table des matières	v
Liste des figures	ix
Liste des tableaux	xiii
Introduction	1
1 Contexte et problématique	5
1.1 Contexte	5
1.1.1 Une révolution pédagogique en marche	5
1.1.2 Le serious game	6
1.1.3 Le business game	6
1.1.4 Le déploiement d'une session formation	8
1.1.5 Le BeerGame	9
1.1.6 Le business game UpStrat	11
1.1.7 Ce qui différencie le BeerGame d'UpStrat	17
1.2 Présentation de la problématique	18
1.2.1 Enjeux	18
1.2.2 Les verrous	18
1.2.3 Les objectifs de la thèse	20
1.3 Conclusion	21
2 Méthodologie et modélisation	23
2.1 Notions de stratégie	23
2.1.1 La démarche stratégique classique	23
2.1.2 Les Facteurs Clés de Succès	24
2.1.3 Définir un DAS : Domaine d'Activité Stratégique	25
2.2 Notions de comptabilité de gestion	27
2.2.1 Définitions et nomenclature	27
2.2.2 Méthode des coûts complets	28
2.2.3 La division de l'activité en produits distincts	29
2.3 La décision du joueur	31
2.3.1 Répartition des rôles	31
2.3.2 L'entreprise, un ensemble de processus en interaction	31
2.3.3 Relations au sein de l'entreprise	33
2.3.4 Le joueur et sa solution	33
2.4 Méthodologie adoptée par l'agent	36
2.4.1 La construction d'une solution complète	36
2.4.2 Proposition d'une approche de résolution par méthode hybride	39

2.4.3	Le tournoi comme outil d'évaluation	40
2.5	Modélisation des départements Production, Distribution et Finance	41
2.5.1	Notations	41
2.5.2	Le département Production	41
2.5.3	Modélisation du département Distribution	49
2.5.4	Modélisation du département Finance	49
2.5.5	Mise en œuvre	53
2.6	Conclusions	54
3	Génération des hypothèses par méthode d'optimisation	57
3.1	Introduction à l'optimisation combinatoire	58
3.1.1	Définition d'un problème d'optimisation combinatoire	58
3.1.2	Modéliser un problème d'optimisation combinatoire	58
3.1.3	Principales méthodes d'optimisation	59
3.1.4	Principaux concepts des méta-heuristiques	60
3.1.5	Méthodes de recherche locale	61
3.2	Méthode d'optimisation proposée	63
3.2.1	Recherche d'un bon ensemble de décisions : un problème d'optimisation combinatoire?	63
3.2.2	Représentation d'une solution	64
3.2.3	Un voisinage variable	65
3.2.4	Une procédure itérative	66
3.2.5	Évaluation d'une solution	66
3.3	Expérimentations	67
3.3.1	Protocole expérimental	68
3.3.2	Comparaison des opérateurs de voisinage	69
3.3.3	Analyse de l'opérateur de perturbation	70
3.3.4	Impact du nombre total d'évaluations	72
3.3.5	Performance de la méthode	73
3.4	Conclusions	74
4	Analyse du comportement des joueurs pour simuler la loi de marché	77
4.1	Approche par la théorie des jeux	78
4.1.1	Qu'est-ce qu'un jeu?	78
4.1.2	Le dilemme du prisonnier	78
4.1.3	Les jeux dynamiques	79
4.1.4	Les jeux coopératifs	80
4.2	Transposition de la théorie des jeux au business game	80
4.2.1	Théorie des jeux et business game	81
4.2.2	Les particularités des business games	81
4.3	Notre approche	82
4.4	Le simulateur FirStrat	84
4.4.1	Présentation générale	84
4.4.2	Présentation des données traitées	84
4.4.3	Difficultés à surmonter	85
4.5	Préparation des données	85
4.5.1	Collecte et extraction des données	85
4.5.2	Linéarisation des données	86
4.5.3	Sélection des données	87
4.5.4	Pré-traitement des données	87

4.5.5	Centrage des données	88
4.6	Datamining : Partitionnement des équipes par clustering	88
4.6.1	Description de l'opération de clustering	90
4.6.2	Qualité d'un clustering	92
4.6.3	Mise en œuvre sur FirStrat	95
4.7	Classification des équipes	100
4.7.1	Généralités sur la classification	100
4.7.2	Phase 1 - Construction d'un classifieur	101
4.7.3	Phase 2 - Classification	102
4.7.4	Évaluation de la performance de la classification	104
4.7.5	Classification sur les données de FirStrat	105
4.8	Prédiction des décisions des équipes : Arbres d'anticipation	107
4.8.1	Les arbres d'anticipation	108
4.8.2	Protocole d'expérimentation des arbres d'anticipation	111
4.8.3	Validation de la construction de l'arbre	111
4.8.4	Expérimentations	112
4.9	Conclusions	115
5	Évaluation de l'approche sur UpStrat	117
5.1	Extraction de connaissances sur UpStrat	118
5.1.1	Présentation des données	118
5.1.2	Segmentation des données par produit	119
5.2	Analyse des comportements	120
5.2.1	Clustering	120
5.2.2	Description des classes de joueurs	123
5.2.3	Classification	128
5.2.4	Arbre d'anticipation	131
5.3	Évaluation du processus global d'optimisation	135
5.3.1	Optimisation sans apport de connaissances	135
5.3.2	Optimisation avec apport de connaissances	139
5.4	Conclusions sur les expérimentations	142
	Conclusion et perspectives	144
A	Modélisation du département Distribution	I
A.1	Programme linéaire et contraintes du département Distribution	I
A.1.1	Notation	I
A.1.2	Contraintes et modèle de programmation linéaire	II
A.1.3	La fonction objectif	III

Liste des figures

1.1	Cycle d'une décision	7
1.2	Contenu d'une session de business game	8
1.3	Organisation d'une session BeerGame (FORRESTER et RIEMER [2007])	10
1.4	Organisation d'une équipe dans le BeerGame (FORRESTER et RIEMER [2007])	10
1.5	Interface principale du simulateur UpStrat	12
1.6	Menu du simulateur UpStrat	12
1.7	Ventes prévisionnelles	13
1.8	Saisie des décisions prévisionnelles	13
1.9	Saisie des décisions Ressources Humaines	13
1.10	Saisie des décisions de Recherche et Développement	14
1.11	Saisie des actions publicitaires	14
1.12	Saisie des politiques tarifaires	14
1.13	Saisie des points de vente	14
1.14	Gestion des outils de production	15
1.15	Saisie des finitions et options	15
1.16	Saisie des décisions financières	15
1.17	Projection des résultats	16
1.18	Calcul du score	17
2.1	La démarche stratégique classique (DEMIL et collab. [2013])	24
2.2	Décomposition d'une entreprise en DAS	26
2.3	Modèle ABELL (ABELL [1980])	26
2.4	Exemple de décomposition stratégique d'un groupe chimique (GARIBALDI [2007])	27
2.5	Arbitrage sur la décomposition de l'activité	29
2.6	Représentation d'un processus	32
2.7	Modélisation d'une entreprise en départements principaux	33
2.8	Construction d'une solution par un joueur	35
2.9	Modèle décisionnel d'un comité de direction	36
2.10	Construction d'une solution par un agent	37
2.11	Génération d'une solution	38
2.12	Parallélisation des optimisations	54
3.1	Classement des principales méthodes d'optimisation	59
3.2	Perturbation lors d'une recherche locale itérée	63
3.3	Représentation du vecteur solution pour un produit p sur un marché z	64
3.4	Mode de génération des voisins	65
3.5	Mode de sélection de solution	67
3.6	Box plot de la stratégie de voisinage	70
3.7	Box plot de l'opérateur de perturbation	71

3.8	Rang obtenu par variation du nombre maximal d'évaluations	72
3.9	Classements à l'issu des remplacements successifs	74
4.1	Génération d'une solution	77
4.2	Jeu simple sous forme extensive	80
4.3	Méthodologie générale	83
4.4	Données brutes extraites	86
4.5	Linéarisation des décisions	87
4.6	Fenêtre de sélection de variables	87
4.7	Détail de la première phase de l'approche	89
4.8	Exemple de clustering	90
4.9	Exemple de dendrogramme	92
4.10	Représentation de cluster sur FirStrat	98
4.11	Positionnement des comportements de joueurs dans la matrice de M. Porter	99
4.12	Méthodologie générale	100
4.13	Classification	101
4.14	Classification par K-NN	103
4.15	Principe global de la validation de la classification	104
4.16	Principe de la validation croisée	105
4.17	Méthodologie générale	108
4.18	Principe de l'arbre d'anticipation	109
4.19	Exemple d'arbre d'anticipation	110
4.20	Représentation de la morphologie des arbres en fonction du choix de l'opérateur de branchement	112
4.21	Provenance des paramétrages dans le processus d'optimisation	113
5.1	Environnement d'une simulation	117
5.2	Méthodologie générale	120
5.3	Représentation des effectifs par cluster pour le produit 1	123
5.4	Représentation des variables décisionnelles associées au produit 1 par cluster sur Upstrat	124
5.5	Représentation des effectifs par cluster pour le produit 2	125
5.6	Représentation des variables décisionnelles associées au produit 2 par cluster sur Upstrat	127
5.7	Représentation des effectifs par cluster pour le produit 2	128
5.8	Représentation des variables décisionnelles associées au produit 3 par cluster sur Upstrat	129
5.9	Rang obtenu après remplacement de la première équipe (sans apport de connaissances)	136
5.10	Rang obtenu après remplacement de la deuxième équipe (sans apport de connaissances)	137
5.11	Rang obtenu après remplacement de la troisième équipe (sans apport de connaissances)	138
5.12	Rang obtenu après remplacement de la quatrième et de la cinquième équipe (sans apport de connaissances)	138
5.13	Rang obtenu après remplacement de la première équipe (avec apport de connaissances)	140
5.14	Rang obtenu après remplacement de la deuxième équipe (avec apport de connaissances)	141

5.15 Rang obtenu après remplacement de la troisième équipe (avec apport de connaissances) 141

Liste des tableaux

2.1	Matrice d'analyse MOFf pour un DAS (BRESSY [2014])	24
2.2	Typologie des coûts dans UpStrat	28
2.3	Exemple de tableau des coûts UpStrat pour une équipe en cours de jeu . . .	30
2.4	Répartition des coûts sur UpStrat pour un exemple de jeu	30
2.5	Ratios de pilotage par fonction opérationnelle	32
2.6	Description d'une solution	34
2.7	Notations générales	41
2.8	Paramètres de production	43
2.9	Variables de production pour un produit p au tour n	43
2.10	Variable des machines	44
2.11	Coûts intégrés dans le programme linéaire de Production	46
2.12	Coûts intégrés dans le programme linéaire de Distribution	50
3.1	Stratégie de voisinage - Statistiques sur douze exécutions	69
3.2	Opérateur de perturbation - Statistiques sur 12 exécutions	71
3.3	Nombre d'évaluations - Statistiques sur 12 exécutions	72
3.4	Paramètres retenus pour la recherche locale	73
3.5	Evaluation de la performance après remplacement des équipes	74
4.1	Matrice des gains : Dilemme du prisonnier	79
4.2	Comparaison des clustering par fenêtre de décisions	96
4.3	Effectif par cluster - Partitionnement en 3 clusters sur la fenêtre 5-8	97
4.4	Comparaison des méthodes de classification	106
4.5	Taux d'équipes bien classées pour les variables sélectionnées	107
4.6	Comparaison des paramètres : centroïd vs. moyenne générale	114
4.7	Comparaison des paramètres des arbres par rapport à la méthodes des centroïds	114
4.8	Comparaison des paramètres des arbres par rapport à la moyenne générale	114
5.1	Choix du clustering - UpStrat	122
5.2	Caractérisation des méthodes de clustering pour chaque produit	123
5.3	Synthèse des positionnements des joueurs par cluster pour le produit 1 . . .	125
5.4	Synthèse des positionnements des joueurs par cluster pour le produit 2 . . .	126
5.5	Synthèse des positionnements des joueurs par cluster pour le produit 3 . . .	128
5.6	Taux de biens classés par méthode de classification	130
5.7	Taux d'équipes bien classées avec les données partielles	131
5.8	Comparaison des paramètres : centroïd vs. moyenne générale	132
5.9	Comparaison des paramètres des arbres par rapport à la méthode des centroïds	133

5.10	Comparaison des paramètres : Arbres d'anticipation paramétrés vs. Moyenne générale	134
5.11	Répartition des rangs obtenus après remplacement de la première équipe (sans apport de connaissance)	136
5.12	Répartition des rangs obtenus après remplacement de la seconde équipe (sans apport de connaissances)	137
5.13	Répartition des rangs obtenus après remplacement de la troisième équipe (sans apport de connaissances)	137
5.14	Répartition des rangs obtenus après remplacement de la quatrième équipe (sans apport de connaissances)	139
5.15	Répartition des rangs obtenus après remplacement de la cinquième équipe (sans apport de connaissances)	139
5.16	Répartition des rangs obtenus après remplacement de la première équipe (avec apport de connaissances)	140
5.17	Répartition des rangs obtenus après remplacement de la deuxième équipe (avec apport de connaissances)	141
5.18	Répartition des rangs obtenus après remplacement du troisième (avec apport de connaissances)	141
5.19	Répartition des rangs obtenus après remplacement de la quatrième équipe (avec apport de connaissances)	142
5.20	Répartition des rangs obtenus après remplacement de la cinquième équipe (avec apport de connaissances)	142
5.21	Préconisation de l'outil d'anticipation de décisions concurrentielles	144
A.1	Paramètres de distribution	I
A.2	Variables de distribution pour une zone z au tour n	II
A.3	Variable des points de vente pour le tour n	II
A.4	Coûts intégrés dans le programme linéaire de Distribution	III

Introduction

Cette thèse s'inscrit dans le domaine de l'optimisation et de l'extraction de connaissances dans un contexte économique et concurrentiel dans le cadre des *business games*. Ce qui fait la principale originalité de notre approche, qui présente un caractère combinatoire, est la méthode de résolution proposée dans le cadre de l'optimisation de décisions au sein d'un simulateur de gestion d'entreprise. Ce type de problème consiste, dans un espace discret de solutions réalisables, à trouver la meilleure solution possible (ou l'ensemble des meilleures solutions). La notion de meilleur est donnée, en général, par un critère de qualité qui peut alors être utilisé comme fonction objectif par une méthode d'optimisation. Lorsque ces problèmes d'optimisation combinatoire sont NP-difficiles, les méthodes exactes demeurent inefficaces pour les résoudre rapidement lorsqu'ils sont de grandes tailles. Afin de résoudre de tels problèmes, il est possible de faire appel aux méthodes heuristiques et en particulier aux métaheuristiques qui sont des méthodes génériques de résolution de problèmes d'optimisation, connues et utilisées pour leur efficacité. Ces métaheuristiques peuvent être classées en deux catégories : les recherches locales, méthodes à solution unique, et les méthodes à base de population. Dans ce document, nous nous concentrerons sur les méthodes à solution unique qui sont sensibles au paysage associé au problème à résoudre. Le paysage d'un problème d'optimisation combinatoire est défini à partir d'une relation de voisinage entre solutions réalisables et d'un critère de qualité de ces solutions, donné, la plupart du temps, par la fonction objectif. Nous verrons dans ce document les difficultés rencontrées pour évaluer la qualité d'une solution pour notre cas d'étude.

Cette thèse est financée par la société Strat&Logic. Elle a été menée au sein de l'équipe DOLPHIN du laboratoire CRISAL de Lille (CNRS, Université de Lille). Cette équipe est commune avec le centre de recherche INRIA Lille-Nord Europe. Elle a pour but de modéliser et de résoudre des problèmes d'optimisation combinatoire (multiobjectif) de grande taille, en utilisant des techniques hybrides, de parallélisation et de calcul distribué. Les thématiques de l'équipe portent sur la mise en œuvre conjointe des méthodes avancées issues de l'optimisation combinatoire en recherche opérationnelle, de la décision en intelligence artificielle. Pour sa part, Strat&Logic, créée en 2000, réalise des produits pédagogiques immersifs adaptés à la formation initiale et professionnelle. La société dispose d'une gamme fournie de produits qui appartient à la famille des *business games*. Les *business games*, encore appelés *simulation de gestion d'entreprise*, sont des simulateurs économiques dans lesquels les participants, répartis en équipes, dirigent des centres d'activités sous contraintes. La présence du mot « game » indique une dimension ludique et compétitive. Les équipes sont mises en compétition et tenteront d'atteindre leurs objectifs respectifs. Les objectifs des différentes équipes seront évidemment incompatibles. En effet, il ne pourra y avoir qu'un seul vainqueur.

Compte tenu de la complexité relative au contexte inhérent aux travaux de cette thèse, le Chapitre 1 est dédié à l'exposition de ce contexte. Dans cette introduction, nous donnons cependant les motivations et objectifs de ce travail, et présentons l'organisation gé-

nérale du document.

Motivations et objectifs

Les générations évoluent. Les modes d'apprentissages doivent s'adapter aux nouvelles générations. L'évolution technologique offre de nouvelles possibilités et parfois une proposition pour relever ce challenge. Elle permet à chacun d'adapter sa formation à son besoin, à son rythme et à sa situation géographique limitant ainsi les contraintes. La formation ou l'enseignement ne sont plus systématiquement associés à la notion de cours magistral où l'enseignant présente de façon descendante ses connaissances à un public de plus en plus hermétique. La formation et l'action doivent désormais se conjuguer. L'apprenant ne peut plus se contenter d'enregistrer, de prendre des notes sans interagir avec l'enseignant. Mieux, celui-ci devient un coach et un accompagnateur du savoir. Il est de plus en plus aux côtés des apprenants et doit quitter son estrade pour se retrouver d'égal à égal avec son public. C'est d'ailleurs ce public qui rythme les enseignements, les rend vivants, nouveaux et ainsi attractifs pour l'enseignant et pour l'assemblée.

Le fort développement et la large diffusion des pédagogies innovantes (classes inversées, pecha kucha, blended learning, apprentissage par problème...) a incité la société Strat&Logic à améliorer ses produits en fluidifiant leur exploitation. Les business games, pour prendre place dans un cursus, sont joués en équipes et imposent une compétition entre elles. Ainsi, le groupe doit disposer d'une taille minimale. Il est, à l'heure actuelle, impossible de jouer seul sur un univers économique. De plus, l'animation de ce type d'activité peut paraître fastidieuse pour un formateur. En effet, ses rôles seront multiples. Il devra, lors de chaque rencontre avec ses apprenants, s'adapter à leur contexte, écouter leurs besoins et proposer des solutions ou voies de recherche permettant de susciter l'interrogation et donc de favoriser l'apprentissage.

Ces observations nous ont amenés à réfléchir à l'opportunité de proposer aux participants un système d'assistance à la prise de décisions qui s'adapterait au contexte du participant et à l'environnement économique dans lequel il est plongé. Ce système pourrait guider les joueurs lors de leurs prises de décisions. Il pourrait également assister l'animateur lors des rencontres avec chaque équipe. Nous avons également imaginé le développement d'un système qui serait concurrent aux joueurs et qui viendrait se mesurer à eux. Ce système pourrait alors se substituer à de vraies équipes et permettrait de gommer la contrainte liée au nombre minimal de participants à une simulation.

Ainsi, l'objectif de cette thèse est d'optimiser des décisions économiques concurrentielles dans un business game. Nous proposons de construire des agents décisionnels qui répondront aux besoins précédemment énoncés. Notre travail sera centré sur UpStrat, business game complet dont l'objectif est d'acquérir les bases du management stratégique d'une PME. Nous proposerons dans ce document une hybridation de méthodes de recherche opérationnelle sur une problématique pédagogique centrée sur les sciences de gestion. Ainsi, au-delà des traitements réalisés sur le simulateur UpStrat, nous tenterons de proposer une démarche généralisable à d'autres produits pédagogiques du même type.

Plan et contributions

Cette thèse est organisée selon le plan suivant.

Dans le **chapitre 1**, nous présenterons en détail le contexte afférent aux travaux de la thèse. Ainsi, nous parcourrons l'évolution de la pédagogie. Nous ferons en particulier un focus sur les serious games avant de mettre en lumière les business games. Nous avons également souhaité présenter un produit très présent dans la littérature : le BeerGame. Nous établirons les singularités du produit sélectionné pour cette étude, UpStrat par rapport au BeerGame.

De plus, nous nous attarderons sur une spécificité de ce cas d'étude. En effet, dans la plupart des problèmes d'optimisation, il est possible de comparer différentes approches de résolution d'un même problème. Cette comparaison est soutenue par un indicateur de performance offert par la fonction objectif. Dans notre étude, nous sommes dans une situation dans laquelle il n'y a pas de fonction objectif. La qualité d'une solution dépend du contexte (des décisions des autres joueurs). Les caractéristiques des solutions générées, exprimées par un ensemble de modalités, sont également originales. Nous présenterons donc ces spécificités. Enfin, nous montrerons que la contrainte de temps d'exécution est très présente dans nos propositions.

Dans le **chapitre 2**, nous présenterons notre méthodologie et la modélisation retenue. Comme nous l'expliquerons, notre originalité est de considérer ce problème de décision dans un business game, comme un problème d'optimisation combinatoire, dont il va falloir préciser les différents éléments (espace des solutions, évaluation des solutions...) en exposant notre modélisation. Cependant, avant de modéliser, nous rappellerons quelques éléments de stratégie et de contrôle de gestion. Par la suite, nous détaillerons les interactions présentes au sein d'une équipe de joueurs. Nous identifierons alors les rôles importants présents dans cette équipe. Fort de ce constat, il deviendra alors possible de décomposer les décisions d'une équipe afin de modéliser indépendamment chaque fonction opérationnelle. Nous proposerons ainsi une transposition réaliste et adaptable de cette décomposition à un agent décisionnel autonome. Nous concluons ce chapitre par l'analyse des principaux départements de l'entreprise que nous avons modélisés au moyen de programmes linéaires en nombre entiers (PLNE). A l'issue du Chapitre 2, nous disposerons de tous les éléments nécessaires à l'optimisation des décisions économiques internes à l'entreprise simulée en disposant de commandes prévisionnelles.

Dans le **chapitre 3**, nous présenterons la méthode d'optimisation que nous proposons pour résoudre le problème ainsi modélisé. Cette approche basée sur une recherche locale aura pour but de définir un ensemble de décisions permettant d'offrir à l'entreprise simulée les meilleurs résultats possibles dans un contexte connu (hors réaction concurrentielle). Afin de positionner notre approche, nous définirons, dans un premier temps, les notions de problème d'optimisation combinatoire et les principales méthodes d'optimisation adaptables à ces problèmes. Nous reviendrons à notre cas d'étude et justifierons notre transposition du problème à une situation de problème d'optimisation combinatoire. Nous définirons alors la notion de *solution* et de *voisinage*. Nous en présenterons leur génération. Nous développerons également dans ce chapitre un mode original de sélection des meilleures solutions afin de pallier l'absence de fonction objectif classique. En fin de chapitre, nous expérimenterons nos choix. Nous tenterons de définir un ensemble de paramètres nous permettant de parcourir efficacement le voisinage. Nous concluons

ce chapitre par les résultats obtenus sur ces développements isolés.

La limite principale de la méthode proposée au chapitre 3 est de ne pas tenir compte de l'environnement concurrentiel, c'est-à-dire de la réaction des concurrents. Or, connaître les décisions des concurrents est nécessaire pour calculer les commandes prévisionnelles qui sont à la base du calcul des performances économiques. Aussi, le but du **chapitre 4** est de fournir une méthode nous permettant d'anticiper les décisions des autres joueurs. Si nous parvenons à anticiper ces réactions, nous évaluerons plus précisément les commandes prévisionnelles de notre agent et assurerons sa performance.

Nous explorerons donc naturellement la théorie des jeux et évaluerons son adaptabilité à notre étude. Par la suite, nous proposerons de nous concentrer sur une méthode d'extraction de connaissances. Nous présenterons donc les méthodes de clustering et de classification. Nous proposerons également un outil original d'anticipation de décisions pour une classe de joueurs : les *arbres d'anticipation*. Ces différentes propositions seront éprouvées par des expérimentations sur un simulateur plus simple : FirStrat qui nous permettra d'en évaluer les limites et opportunités.

Dans le **chapitre 5**, nous revenons au sujet de notre cas d'étude : UpStrat. Ainsi, ce chapitre sera consacré aux expérimentations liées au déploiement de notre méthodologie d'extraction de connaissances sur ce produit plus exigeant. Dans un second temps, nous évaluerons la pertinence d'intégrer de la connaissance pendant la phase d'optimisation. Nous testerons alors notre processus d'optimisation globale d'une part sans intégration de connaissances et d'autre part avec intégration de connaissances. En fin de chapitre, nous concluons sur nos expérimentations et formulerons quelques perspectives.

Un dernier chapitre de conclusion et perspectives reprendra l'ensemble de nos contributions et ouvrira sur différentes perspectives, que ce soit en terme d'amélioration ou de généralisation de la méthode proposée.

Chapitre 1

Contexte et problématique

L'objectif de ce chapitre est de fixer le cadre de l'étude. Ainsi, nous avons souhaité commencer par un aperçu des pratiques pédagogiques efficaces. Cette découverte nous approchera pas à pas du monde des *business games*. De plus, nous définirons les concepts et terminologies de ces activités pédagogiques. En fin de chapitre, nous présenterons la démarche et la finalité du travail de recherche effectué. Enfin, nous positionnerons ce manuscrit dans l'environnement scientifique connu.

1.1 Contexte

1.1.1 Une révolution pédagogique en marche

Le monde a quitté l'ère industrielle pour entrer dans un environnement post-industriel : l'économie de la connaissance. Cette nouvelle économie repose sur : « la production et la distribution de savoirs et d'informations, plutôt que la production et la distribution de choses » (DRUCKER et DRUCKER [1994]). La création de connaissances et leur utilisation sous toutes leurs formes sont désormais au cœur de la création de valeur des entreprises, constituant le troisième facteur de production des entreprises, la créativité humaine devenant le cœur même de nos sociétés (FORAY [2010]).

Face à cette évolution, les sciences de l'éducation évoluent (BEREITER [2005], HARGREAVES [2003]). En effet, le modèle éducatif standard n'est plus adapté à son public (MACHUCA [2000], LEHTINEN [2002], BAKER et O'NEIL [2002]). Malheureusement, l'enseignement dispensé aujourd'hui est trop souvent standardisé, avec des modules hétérogènes, avec comme seules sources de connaissances : l'enseignant et les manuels, un apprentissage reposant sur le mécanisme de mémorisation : transmission/acquisition, et des évaluations individuelles faisant appel à cette même mémorisation (SAWYER [2008]).

Au contraire, le système éducatif ne formera les managers de l'économie de la connaissance qu'avec des modèles pédagogiques alternatifs. Au-delà d'une modification des programmes, c'est en rendant les apprenants acteurs de leur propre formation que le déficit sera relevé. Au travers de leur implication dans une éducation transversale et active avec des sources d'information variées et au sein de groupes collaboratifs, les apprenants atteindront leur objectif. Les méthodes pédagogiques du courant pédagogique dit de « l'éducation nouvelle » sont particulièrement adaptées au développement des besoins actuels des apprenants. Elles défendent le principe d'une participation active des individus à leur propre formation en s'appuyant sur la pédagogie active et la confiance dans les ressources propres à chacun. Elle prône un apprentissage à partir du réel et du libre choix des activités : jeux de rôle, apprentissage par problème...

1.1.2 Le serious game

Etienne Armand AMATO (AMATO [2007]) définit le serious game comme un jeu vidéo permettant aux participants de progresser, de s'adapter dans un environnement, de comprendre des phénomènes ou d'adhérer à des messages publicitaires, idéologiques ou encore politiques ...

Julien ALVAREZ (ALVAREZ [2007]) rejoint assez largement le précédent avis et conclut que le serious game est une application informatique dont le but est de combiner des aspects sérieux comme la formation, la communication, l'apprentissage tout en y intégrant un aspect ludique. L'idée principale de ces outils est donc d'aller bien plus loin que le simple divertissement tout en exploitant quelques composantes importantes : scénario, environnement, classement, récompense ...

Ainsi, comme Honda MOUAHEB (MOUAHEB et collab. [2012]), nous pouvons dresser une carte d'identité des principales caractéristiques des *serious games* :

- Le principal objectif du serious game est d'être un outil d'enseignement.
- Le serious game présente un aspect divertissant.
- Le serious game est un outil informatique souvent assimilable à un jeu vidéo.
- Le serious game présente de multiples objectifs de formation : enseigner, former, éduquer, soigner, entraîner.
- Le serious game est présent dans de nombreux domaines : éducation, santé, défense, politique, publicité, économie.
- Le serious game s'adresse à tout le monde sans restriction d'âge.

1.1.3 Le business game

Le business game (encore appelé « la simulation de gestion » ou « jeu d'entreprise ») fait partie de la catégorie des serious games. Ce type d'activité, basée sur la simulation, occupe une place importante dans les nouvelles pratiques d'apprentissage (KRIZ [2008]).

Le Business game est un outil permettant la découverte et l'utilisation de connaissances et surtout le développement des savoir-être liés à l'entreprise, de manière concrète, ludique et interactive (JOHNSON et collab. [1998]). Il permet aux participants de vivre une expérience d'entreprise, dans une mise en situation réelle d'un environnement complexe et un apprentissage autonome et interactif dans le cadre d'un travail collaboratif.

Les Business games sont des outils informatisés qui modélisent l'environnement concurrentiel et les interactions entre les décisions des entreprises. Ces outils permettent de confronter les décisions des différentes équipes, en y intégrant des paramètres d'environnement économique. Ils génèrent des résultats quantifiés après confrontation : quantification du/des marché(s) (ventes, parts de marché); résultats de chaque équipe (économiques, financiers, marketing, production, humains...); informations sur l'évolution des marchés, attentes des clients ou consommateurs, comportements et décisions des concurrents ...

Les participants sont regroupés en équipes. A chaque équipe, la gestion d'une entreprise virtuelle est confiée. Toutes les équipes sont en concurrence sur un marché. Tous les autres concurrents démarrent dans la même situation avec les mêmes moyens mais pas les mêmes ambitions. Ils doivent ainsi prendre des décisions dans le but de développer au mieux leur activité fictive et d'atteindre les objectifs qu'ils se sont eux-mêmes fixés. Cet aspect concurrentiel impose une motivation et une implication totale des participants à l'action pédagogique. Chaque équipe doit agir pour obtenir la « meilleure » performance.

À partir de données initiales fournies sur cette entreprise (en fonction du scénario choisi) et de son environnement, chaque équipe doit après avoir élaboré un diagnostic stratégique de la situation initiale (concurrence, évolution des marchés, ressources et capacités de l'entreprise), élaborer une stratégie (croissance, développement, pérennité ...) et prendre des décisions opérationnelles sur différentes dimensions : marketing, commercial, finance, production, ressources humaines, logistique, investissement ...

Les informations, extrêmement nombreuses, ne peuvent être assimilées par une seule personne. Cela suppose une organisation voire une spécialisation des participants par domaine. Cette séparation des rôles impose donc de faire confiance aux autres membres de l'équipe. Cette partie du Business game a une influence notable sur le résultat final des équipes. L'objet de la séance de réflexion est de déboucher sur un consensus qui permet de prendre un ensemble de décisions cohérentes malgré le contexte d'incertitude et d'informations imparfaites d'autant plus que les décisions d'une équipe sont contre-carrées par les décisions des autres. Chaque décision a des impacts internes (climat social, niveau de stocks ...) à l'entreprise et des impacts externes sur le marché (obtention de parts de marché, impact sur l'élasticité des marchés ...). Elle est prise, dans chaque équipe, collectivement après une phase de coopération (BARTH et GÉNIAUX [2010]).

L'ensemble de ces décisions est saisi numériquement et une confrontation mathématique produit les résultats (Bilans, comptes de résultats, états de situation, études de marché ...) pour chaque équipe à l'issue d'une période d'activité simulée de l'entreprise. Cette période élémentaire, appelée « tour » est d'une durée variable suivant les applicatifs. Les résultats produits par ces outils sont le fruit de modèles mathématiques intégrant des réalités économiques et de paramètres environnementaux choisis par l'animateur pour l'exercice. Les résultats de la période écoulée sont remis aux participants. Ainsi, un nouveau cycle de décisions peut alors être joué. La figure 1.1 reprend schématiquement le cycle de décision d'une équipe.

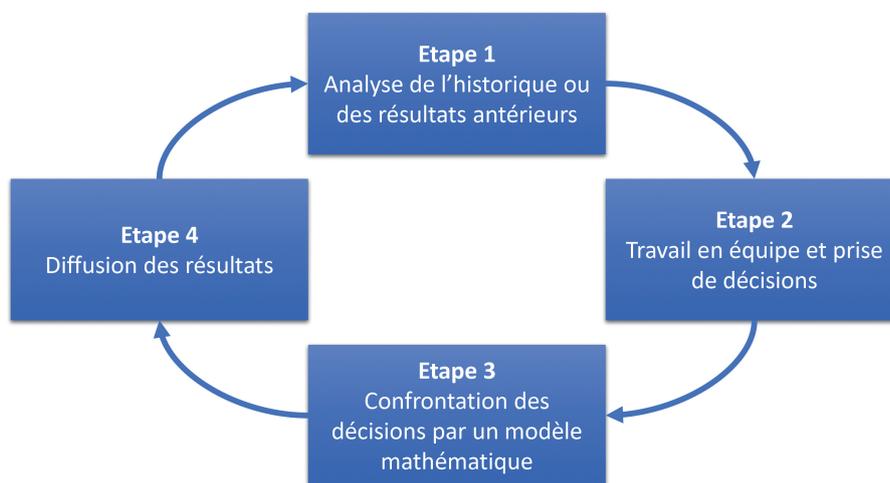


FIGURE 1.1 – Cycle d'une décision

L'animation du business game est faite par un animateur dont le rôle est très central. Non seulement il définit le scénario de départ, mais il maîtrise l'évolution conjoncturelle de l'environnement des entreprises et les paramètres de fonctionnement de ces dernières.

Il peut incarner le rôle de banquier pour l'accord de facilités de caisse ou pour négocier les taux d'intérêt lors du financement d'investissements. Il peut également jouer, selon l'objectif pédagogique, d'autres rôles : représentant du personnel, donneur d'ordres, fournisseur, sous-traitant, consultant, organisme de propriété industrielle et également

expert-comptable.

Il explique les règles du jeu aux étudiants, et tout au long de l'exercice, les aide à comprendre les données, à construire des raisonnements et rappelle des éléments techniques pour analyser et décider. Il donne au business game un rythme, notamment le temps alloué pour chacune des périodes. Les apprenants sont ainsi immergés dans le contexte et décident sous contraintes et en temps limité, nécessairement court.

Ces différentes missions imposent à l'animateur d'être formé au jeu, d'être pluridisciplinaire et de disposer d'une expérience probante du monde économique.

Comme le présente la figure 1.2, une session pédagogique basée sur un business game comprend trois éléments fondamentaux : le simulateur, le scénario pédagogique et l'animation.

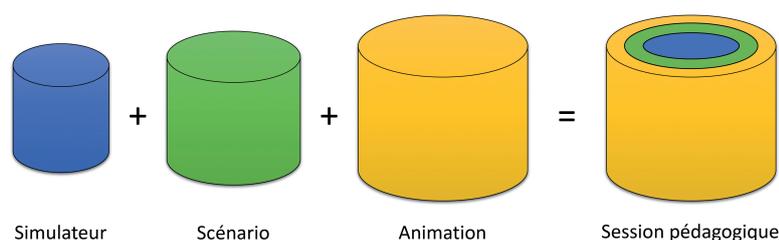


FIGURE 1.2 – Contenu d'une session de business game

Les business games sont régulièrement utilisés dans les formations supérieures en tant qu'outils d'apprentissage pour la gestion, la stratégie, la finance... Aussi, il existe différents niveaux de business games. Nous trouvons par exemple des jeux d'introduction à la gestion généralistes et simples. Ils mettent en œuvre un petit nombre de variables et permettent d'aborder les principales problématiques de la gestion d'une entreprise. Nous trouvons aussi des jeux d'approfondissement : ils développent une dimension spécifique, par exemple le marketing, ou les aspects comptables de la gestion d'une entreprise. Enfin, il existe des jeux d'immersion qui multiplient les variables de manière à plonger les participants dans un univers complexe qui est censé représenter la « vraie vie » de l'entreprise.

1.1.4 Le déploiement d'une session formation

Le déploiement de session de formation exploitant des business games a longtemps été considéré comme étant lourd. En effet, il est nécessaire de planifier l'exploitation de ressources synchronisées : participants, enseignants et matériel informatique. Depuis quelques années, Internet est accessible dans tous les organismes. Il n'est donc plus indispensable de monopoliser des salles informatiques pour déployer des formations de ce type. L'exploitation de tous types de terminaux suffit à la bonne exploitation des produits. Le développement des pratiques de formation en « blended » (une partie de la formation se fait en salle et l'autre partie à distance) tend aussi à se développer grâce aux moyens technologiques modernes : plate-formes de formation, smartphones, applications spécifiques externes... Ces évolutions lèvent donc naturellement le frein technique.

En terme d'encadrement, une session, pour être pédagogiquement efficace, doit être encadrée correctement. Ce niveau d'encadrement doit s'adapter aux profils des apprenants. Ainsi, par expérience, la taille des groupes qu'un animateur est susceptible d'encadrer varie classiquement de quinze à quarante personnes. Dans un contexte de cohésion de groupe, où les besoins pédagogiques et les sollicitations à l'animateur sont réduites, un gros effectif peut convenir. Lorsque les stagiaires imposent beaucoup d'interactions et

requiert une aide technique intense, le recours à un taux d'encadrement bien plus élevé sera opportun.

Cet encadrement est fondamental. Aussi, il n'est pas concevable de demander à un animateur de réaliser en direct une analyse pour un nombre trop important d'équipes. Ceci est renforcé si la complexité de la simulation est élevée. En situation normale, l'expérience prouve qu'un animateur, même expert, ne peut superviser correctement plus de six équipes.

Comme indiqué précédemment, l'animateur sera :

- **pédagogue** : il reprendra avec les équipes les principales théories qui régissent l'économie,
- **régulateur** : il sera le garant des règles du jeu et du tempo de la formation,
- **acteur** : il incarnera l'environnement de l'entreprise et ses parties prenantes.

Il n'est pas toujours aisé de trouver ce type de profil dans tous les établissements.

Nous venons ainsi de présenter les contraintes maximales imposées par les business games. Cependant, nous pouvons aussi ajouter une contrainte minimale qui réside dans la taille du groupe. En deçà de neuf participants, il est difficile d'organiser un jeu puisque l'intensité concurrentielle disparaîtrait au profit d'un duopole économique.

1.1.5 Le BeerGame

Parmi les business games, le Beer game fait l'objet de nombreuses publications scientifiques. Nous le présenterons ci-après et dresserons en fin de paragraphe les principaux aspects qui le différencient du simulateur sur lequel nous nous consacrerons dans notre étude.

Présentation du BeerGame

Le BeerGame est un jeu de supply chain et de logistique en business to business. Il a été inventé dans les années 1960 par Jay Forrester et présenté au MIT à la suite de son travail sur la dynamique du système (DISNEY et TOWILL [2003]).

Lors d'une session sur le BeerGame, les participants sont répartis en quatre équipes de trois. Les équipes entretiennent des relations de client/fournisseur sur le marché de la bière. La figure 1.3 présente les quatre acteurs d'une partie :

- le **fabricant** (Factory) : en charge de la production des bières,
- le **distributeur** : intermédiaire entre le fabricant et le grossiste,
- le **grossiste** : intermédiaire entre le distributeur et le détaillant,
- le **détaillant** : assimilable à un magasin en contact direct avec le consommateur final.

Chaque table accueille une équipe. Aucune communication n'est autorisée entre les équipes. Toutes les tables sont organisées comme le présente la figure 1.4 :

- La place pour les **commandes entrantes** (*incomming order*) en provenance de l'entreprise en aval
- La place pour les **ordres sortants** (*outgoing order*) en direction de l'entreprise en amont
- La place pour les **réceptions** (*Incoming delivery*) en provenance de l'amont

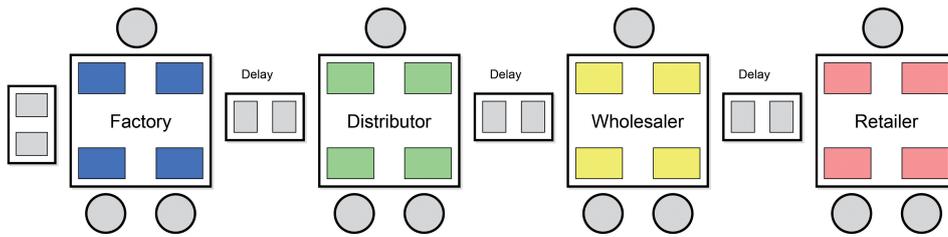


FIGURE 1.3 – Organisation d'une session BeerGame (FORRESTER et RIEMER [2007])

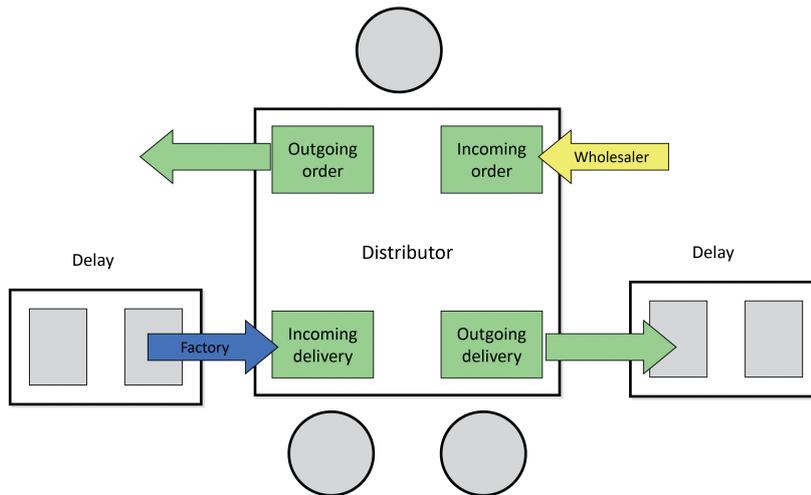


FIGURE 1.4 – Organisation d'une équipe dans le BeerGame (FORRESTER et RIEMER [2007])

— La place pour les **livraisons** (*outgoing deliveries*) en direction de l'aval

La période élémentaire de jeu, un tour, est la semaine. Les livraisons mettent deux semaines à arriver à destination. La production prend également deux semaines.

Le but du jeu est de **minimiser** le coût.

Les coûts sont générés par les produits en stock (0,5 € / semaine et par produit stocké). Ils sont aussi augmentés par les commandes non livrées ou en retard appelées dans le jeu *Backorders* (1 € / semaine de retard et par backorder).

Le principe général de jeu pour chaque acteur de la chaîne de distribution est :

1. Réception des nouvelles livraisons et mise à jour des feuilles de gestion des stocks et commandes,
2. Réception des commandes et mise à jour des feuilles d'expédition,
3. Calcul du montant total à expédier,
4. Décision en équipe sur la commande à passer à l'entreprise en aval.

Le principe du jeu est simple. Il permet de présenter en particulier comment les éléments isolés d'un système s'influencent mutuellement. Il illustre aussi comment la pensée individuelle diffère de la pensée systémique. Enfin, il permet de découvrir et d'introduire des mécanismes d'optimisation des systèmes et l'utilisation des systèmes d'information.

Ce jeu a permis de mettre en évidence le *bullwhip effect* connu en français sous *l'Amplification de la Variabilité de la Demande* (AVD). Il s'agit d'un symptôme bien connu des

problèmes de coordination des chaînes logistiques traditionnelles. Il est causé par des manques de communication entre les différents protagonistes d'une chaîne logistique.

Les nombreux articles qui ont été publiés ont permis de tester l'efficacité pédagogique de la simulation (RIEMER [2008]), d'analyser le phénomène d'AVD (GOODWIN et FRANKLIN [1994]) et de présenter la performance d'agents économiques autonomes qui se sont substitués aux joueurs (KIMBROUGH et collab. [2002] ...).

1.1.6 Le business game UpStrat

UpStrat fait partie de la gamme de business game de la société STRAT&LOGIC [2000]. Il fait l'objet de notre étude. Ainsi, nous avons souhaité présenter ce simulateur afin d'en découvrir les principales règles du jeu et possibilités offertes aux joueurs.

Présentation d'UpStrat

UpStrat est particulièrement adapté à notre étude. Il fait partie des business games « immersifs » par la quantité de décisions à prendre et par sa proximité à la réalité du pilotage stratégique d'une entreprise industrielle. En effet, il illustre les principales fonctions stratégiques d'une entreprise.

Le marché simulé est constitué de produits qui sont distribués sur des marchés. Le croisement de ces données nous permet d'obtenir un ensemble de couples produit / marché encore appelés segments de marché. Pour chaque segment, les participants devront définir leurs objectifs, mettre en place la stratégie commerciale et industrielle la plus adaptée pour atteindre leurs objectifs le plus vite possible. Ils analyseront aussi les indicateurs de performance financière par produit (marges opérationnelles, surcoût de production...).

Tous les paramètres de cette simulation peuvent changer à tout moment en fonction d'événements économiques prédéfinis dans le scénario ou tout simplement en fonction de la volonté de l'animateur. Les principaux paramètres qui sont susceptibles d'évoluer sont : la sensibilité des clients, les impôts et taxes, les coûts externes... .

Pour prendre des décisions éclairées, les participants doivent explorer plusieurs alternatives comme le ferait un comité de direction. Ils doivent ainsi évaluer la pertinence de chacune d'elles afin d'en sélectionner une seule. Le choix de l'alternative la plus performante doit se faire en fonction des résultats prévus pour la période en cours mais aussi et surtout en fonction des positionnements de concurrents et de la stratégie à moyen ou long terme de l'entreprise.

Notons que ce simulateur offre l'opportunité de déployer des stratégies à long terme. Les investissements faits par les équipes ont des impacts sur la décision courante mais aussi sur les décisions à venir. Il est donc opportun de calculer les retours sur investissements des choix et d'en définir les priorités.

Ainsi, la plupart des décisions des équipes ont un impact sur leurs performances internes mais aussi sur le marché par le biais des élasticités de marché et par la réaction des équipes à leurs résultats.

Règles du jeu du simulateur UpStrat

L'objet de ce paragraphe est de présenter concrètement les missions d'une équipe. Lors de la connexion au simulateur, l'équipe découvre le jeu par l'interface générale présentée en figure 1.5.

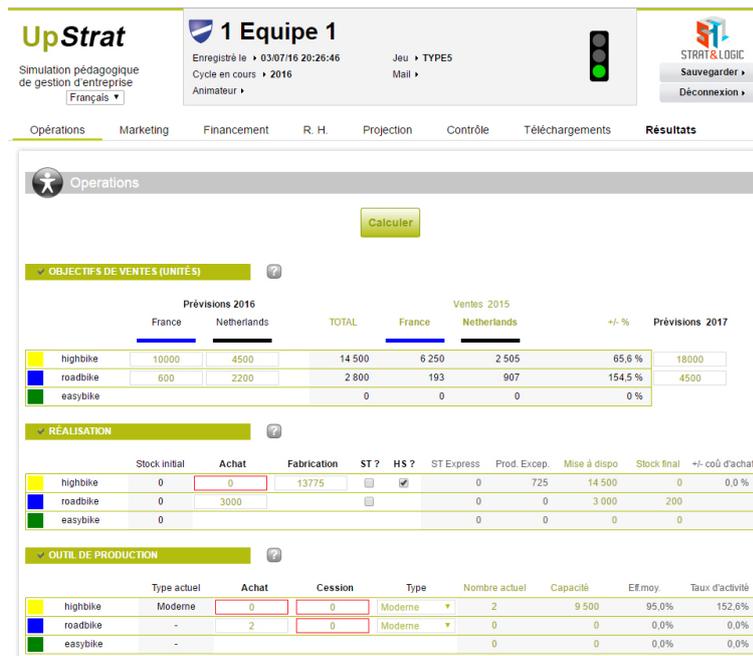


FIGURE 1.5 – Interface principale du simulateur UpStrat

Cette interface permet dans sa partie supérieure de reprendre les principales informations relatives aux jeux en cours : cycle actuel, animateur, numéro d'équipe ...

Notons sur la droite du cartouche un feu tricolore qui permettra aux joueurs d'informer la centrale de jeu et donc l'animateur de son niveau de maturation de décisions. Un feu tricolore positionné sur le rouge indique qu'aucune décision n'est validée. Lorsqu'il est positionné par l'équipe à l'orange, la décision est en cours de validation. Enfin, sur la position verte, le joueur est prêt à passer à l'étape suivante et attend ses résultats.

Le menu à disposition (Figure 1.6) permet de :

- prendre des décisions sur les onglets « Opérations », « Marketing », « Financement », « RH »,
- d'évaluer ses choix sur les onglets « Projections » et « Contrôle »,
- de consulter les résultats des tours passés sur les onglets « Téléchargements », « Résultats ».



FIGURE 1.6 – Menu du simulateur UpStrat

Notons que l'onglet « Projections » offre un expert comptable disponible en temps réel aux joueurs. Pour chaque décision validée, une évaluation financière peut être visionnée.

Les prochaines figures proposées permettent de parcourir les écrans de prise de décisions.

Commençons par la figure 1.7 qui est consacrée à la saisie des prévisions de vente pour le cycle en cours pour les deux marchés ouverts sur ce scénario de marché (France et Pays-Bas) et pour les trois produits finis, des cycles commercialisés sous les noms : *Highbike*, *Roadbike*, *Easybike*.

OBJECTIFS DE VENTES (UNITÉS)							
	Prévisions 2016			Ventes 2015			Prévisions 2017
	France	Netherlands	TOTAL	France	Netherlands	+/- %	
highbike	10000	4500	14 500	6 250	2 505	65,6 %	18000
roadbike	600	2200	2 800	193	907	154,5 %	4500
easybike			0	0	0	0 %	

FIGURE 1.7 – Ventes prévisionnelles

La figure 1.8 permet aux équipes de saisir leurs décisions de production. Elles peuvent également sous traiter une partie du besoin. Les cases à cocher sont nécessaires à la programmation des heures supplémentaires et pour le recours à la sous-traitance express.

RÉALISATION										
	Stock initial	Achat	Fabrication	ST ?	HS ?	ST Express	Prod. Excep.	Mise à dispo	Stock final	+/- coût d'achat
highbike	0	0	13775	<input type="checkbox"/>	<input checked="" type="checkbox"/>	0	725	14 500	0	0,0 %
roadbike	0	3000		<input type="checkbox"/>		0	0	3 000	200	
easybike	0					0	0	0	0	

FIGURE 1.8 – Saisie des décisions prévisionnelles

La figure 1.9 présente les décisions en ressources humaines : l'augmentation des salaires, l'allocation de primes au personnel et le budget alloué à la formation indexé sur la masse salariale de l'entreprise et exprimé en taux.

	Production		Administration	
	2015	2016	2015	2016
Indice de rémunération	101,0	103,0	101,0	103,0
Prime	3,0 %	5,0 %MS	3,0 %	5,0 %MS
Formation (k€)	1,4 %	1,8 %MS	1,5 %	2,0 %MS

FIGURE 1.9 – Saisie des décisions Ressources Humaines

La figure 1.10 permet de visualiser l'interface de saisie réservée à l'enregistrement des efforts de R&D consentis par l'entreprise. Ces efforts sont affichés sur une échelle/grille. A chaque étape de la grille correspondent des coûts qui peuvent être considérés comme étant des charges d'exploitation récurrentes ou ponctuelles ou des investissements. Certains investissements en R&D offrent des apports techniques aux entreprises. Ces apports sont des réductions de temps de production, de consommation de matières premières par exemple.

La figure 1.11, quant à elle, est dédiée à la présentation de la saisie des efforts publicitaires. Le ciblage publicitaire, exprimé sous la forme de modalités, entraîne des coûts et des positionnements concurrentiels qui impactent la répartition du marché.

La politique prix, exprimée par modalités dans ce simulateur, est enregistrée par couple produit/marché comme présenté sur l'écran présenté figure 1.12.

Le type de canal de distribution et son dimensionnement sont enregistrés également comme le montre la figure 1.13. Les saisies entraînent des calculs élémentaires à destination des participants afin de limiter les incompréhensions et maximiser le temps de réflexion et d'arbitrage.

✓ DÉVELOPPEMENT PRODUIT ?

highbike	0	Stage 1	Stage 2	Stage 3	Stage 4	Stage 5
		■	■	■	□	□

FIGURE 1.10 – Saisie des décisions de Recherche et Développement

✓ PUBLICITE ?

	2016		Budget (k€)	
	France	Netherlands	France	Netherlands
highbike	Level 2 (4 % of TO)	None	74	0
roadbike	Level 1 (2 % of TO)	Level 5 (10 % of TO)	13	271
easybike			0	0

FIGURE 1.11 – Saisie des actions publicitaires

✓ PRIX (€) ?

	2016		Prix (€)	
	France	Netherlands	France	Netherlands
highbike	Marché	Marché	185,0	195,0
roadbike	+1	+4	1 071,0	1 230,5
easybike			0,0	0,0

FIGURE 1.12 – Saisie des politiques tarifaires

✓ COMMERCIALISATION / DISTRIBUTION ?

	France	Netherlands
Ancien mode	Succursa...	Succursa...
Mode	Succursales	Succursales
Nombre minimum	2	3
CA prévu	2 493 k€	3 585 k€
Nombre en 2015	1	2
+/- Variation	1	1
= Nombre en 2016	2	3
soit CA maxi	3 000 k€	3 600 k€

FIGURE 1.13 – Saisie des points de vente

Le dimensionnement de l’outil de production est intégré dans la décision (cf. figure 1.14). Chaque équipe devra y faire figurer le type d’outil de production sélectionné pour chaque référence de produit fini. Elle devra également dimensionner sa force productrice et éventuellement procéder au renouvellement des lignes de production les plus âgées.

OUTIL DE PRODUCTION								
Type actuel	Achat	Cession	Type	Nombre actuel	Capacité	Eff.moy.	Taux d'activité	
highbike	0	0	Moderne	2	9 500	95,0%	152,6%	
roadbike	2	0	Moderne	0	0	0,0%	0,0%	
easybike	-	-	-	0	0	0,0%	0,0%	

FIGURE 1.14 – Gestion des outils de production

La finition ou les options proposées aux clients sont des facteurs de différenciation importants. Ces décisions sont prises au moyen de l’interface présentée sur la figure 1.15.

FINITION/OPTIONS					
	2016		Prix (€)		
	France	Netherlands	France	Netherlands	
highbike	Road	Road	3	3	
roadbike	Green	Green	25	25	
easybike					

FIGURE 1.15 – Saisie des finitions et options

Enfin, les décisions financières concernant le recours à l’emprunt, à l’augmentation de capital, à la distribution de dividendes ou tout simplement à la définition des délais de paiement des clients et des fournisseurs sont présentées sur la figure 1.16.

CRÉDIT CLIENT ET FOURNISSEUR					
Clients		France	Netherlands	Fournisseur	
2016	30	30	2015	Jours	
Variation %	0,00 %	0,00 %	Délai de règlement	60	60
			Escompte de règlement		0 k€
OPTIONS DE FINANCEMENT(K€)					
Emprunts à long terme	à taux fixe	MT	LT	Capacité	
	à taux variable	300	0	Emprunts	2 370 k€
		0	0	Capital	360 k€
				Consummé	300 k€
Augmentation de capital	2015	2016	Concours bancaire		
Distribution/action (€)	0	0	Maxi	680 k€	
	0,00	0,50	Utilisé	0 k€	

FIGURE 1.16 – Saisie des décisions financières

Une fois toutes ces décisions saisies, la projection financière présentée sur la figure 1.17 permet de visualiser les résultats prévisionnels de l’équipe. Nous observons ici un compte d’exploitation analytique. Cette projection est enrichie par un bilan prévisionnel et un tableau de trésorerie lui aussi prévisionnel. Quelques ratios financiers viennent compléter l’analyse des joueurs.

La pertinence de choix peut être validée par ces projections. Cependant, celles-ci sont réalisées par rapport aux prévisions de ventes saisies par les entreprises. Ainsi, une prévision de ventes très éloignée de la réalité du marché offrira des projections financières flatteuses ou au contraire pessimistes et conduira les équipes à des déconvenues ou embellies auxquelles les équipes devront s’adapter au cycle suivant.

	2015					2016					Variation %
	highbike	roadbike	easybike	k€	%	highbike	roadbike	easybike	k€	%	
CHIFFRE D'AFFAIRES	1 676	1 270	0	2 946		2 728	3 350	0	6 077		
Réductions sur ventes	0	0	0	0		0	0	0	0		
CHIFFRE D'AFFAIRES	1 676	1 270	0	2 946	100,0%	2 728	3 350	0	6 077	100,0%	106,3%
COUT D'ACHAT	693	594	0	1 287		806	1 620	0	2 426		88,5%
Variation de stock	-40	0	0	-40		0	108	0	108		-373,4%
COUT DES VENTES	732	594	0	1 326	45,0%	806	1 512	0	2 318	38,2%	74,8%
MARGE BRUTE	944	676	0	1 620	55,0%	1 921	1 838	0	3 759	61,8%	132,1%
Frais de développement	80	120	0	200	7	30	80	0	110	1,8%	-45,0%
Frais liés à la force de vente	104	95	0	199	6,8%	175	187	0	362	5,9%	81,8%
Frais de publicité	23	75	0	98	3,3%	74	284	0	358	5,9%	264,4%
Frais de transport	51	33	0	85	2,9%	86	84	0	170	2,8%	100,8%
Frais liés à l'administration	155	103	0	258	8,7%	258	292	0	550	9,1%	113,5%
Frais liés à la différenciation	0	0	0	0	0,0%	44	70	0	114	1,9%	-
Autres charges d'exploitation	0	0	0	0	0,0%	0	0	0	0	0,0%	-
Dotations aux amortissements	35	26	0	61	2,1%	49	61	0	110	1,8%	80,3%
CHARGES D'EXPLOITATION	448	453	0	901	30,6%	715	1 058	0	1 773	29,2%	-
RESULTAT D'EXPLOITATION	496	223	0	719	24,4%	1 206	780	0	1 986	32,7%	-
Charges financières				16	0,5%				32	0,5%	98,6%
Produits financiers				2	0,1%				7	0,1%	320,5%
RESULTAT FINANCIER NET				-15	-0,5%				-25	0,0%	74,4%
RESULTAT COURANT				705	23,9%				1 960	0,0%	178,2%
Résultat exceptionnel				0	0,0%				0	0,0%	-
Participation des salariés				0	0,0%				0	0,0%	-
Impôt sur les sociétés				235	8,0%				653	10,7%	178,2%
RESULTAT NET				470	16,0%				1 307	21,5%	178,2%

FIGURE 1.17 – Projection des résultats

Un mode de classement original

Afin de comparer les équipes dans l'environnement économique, Upstrat intègre un **score**. C'est l'indicateur utilisé pour le classement des équipes.

Il existe de nombreuses méthodes permettant de « scorer » les sociétés (BARDOS [2001] et BARDOS et collab. [2005]). Ces cotations réalisées depuis les années 1990 par la banque de France (par exemple) ont pour but d'anticiper les risques de défaillance des entreprises. Ces risques sont anticipés à partir de ratios financiers passés et prospectifs. L'agglomération de ces ratios offre une note et ainsi une évaluation du risque aux créanciers, actionnaires et investisseurs. Les méthodes d'élaboration de ces indicateurs évoluent régulièrement car elles doivent coller au mieux à la réalité économique.

Dans le cas d'UpStrat et afin d'accroître les écarts entre les entreprises d'un même univers, nous avons souhaité créer notre propre mode de cotation ou classement. Le score est composé des ratios suivants :

- La rentabilité d'exploitation,
- Le taux de livraison,
- La part de marché en chiffre d'affaires,
- Le résultat par action,
- Le dividende par action,
- L'efficacité moyenne du personnel.

Pour chacun des ratios précédemment cités, les équipes reçoivent des points. Ceux-ci sont obtenus en fonction du rang qu'elle atteint pour un tour de jeu sur le ratio sélectionné. Par exemple, l'équipe qui se placera première en part de marché recevra dix points. La seconde recevra six points. Chaque ratio est ensuite pondéré en fonction de paramètres définis dans la simulation. Ces paramètres sont connus en début de simulation et ne varient pas. La somme des points obtenus pour chaque critère donne le score

de l'année de l'équipe (*points bruts*) comme la figure 1.18 le présente. En cas de problème de solvabilité court terme (découvert bancaire) ou long terme (surendettement), des pénalités sur ce score brut sont appliquées. Nous obtenons ainsi le score de l'année.

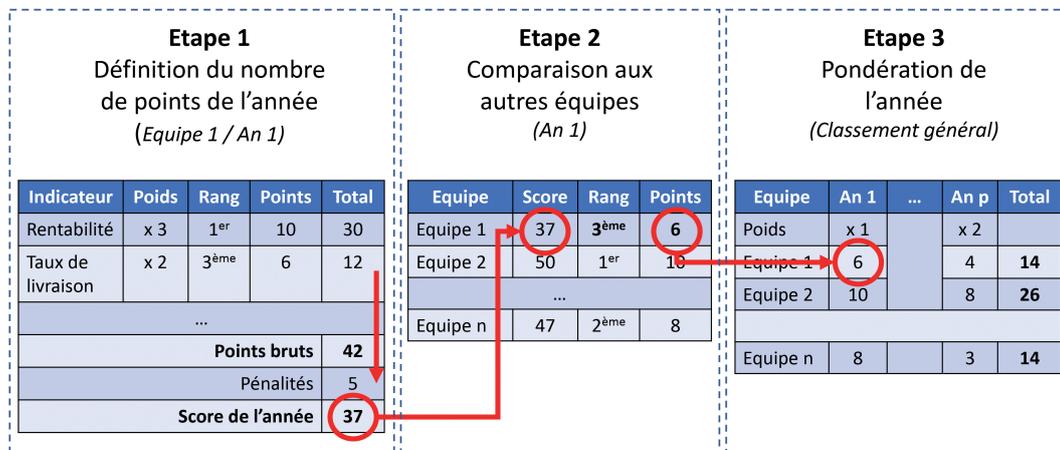


FIGURE 1.18 – Calcul du score

Ce score est ensuite comparé aux scores des autres équipes. L'équipe la mieux placée sur le tour de jeu recevra le maximum de points pour ce tour et ainsi de suite pour les équipes suivantes dans le classement de l'année.

De plus, les tours sont pondérés. Par exemple, en début de simulation, chaque année est coefficientée à 1. Il n'y aura donc, par exemple, que 10 points qui seront distribués pour le premier tour au vainqueur. Habituellement, après quelques tours de jeu, le poids des tours augmente. Cette pondération évolutive permet de récompenser les équipes proposant une stratégie envisagée à long terme. Elle permettra également à toutes les équipes de revenir dans la partie même si leur début de simulation a été difficile.

Le gagnant de la simulation sera l'équipe qui totalisera le plus de points en fin de jeu. Le classement révèle ainsi la performance totale de l'équipe pour tous les tours de jeu réalisés. Il n'est pas le seul reflet du dernier tour.

1.1.7 Ce qui différencie le BeerGame d'UpStrat

Nous venons de présenter successivement deux business game : le BeerGame et UpStrat. Nous pensons qu'il est important de souligner les différences majeures entre ces deux modèles.

Le BeerGame illustre très bien la chaîne logistique entre acteurs économiques. Les équipes ne sont pas concurrentes et entretiennent des relations client/fournisseur sans échange d'informations verbales. Les transactions à l'achat ou à la vente sont les seules liens entre les équipes.

UpStrat n'a pas vocation à illustrer ce type d'échanges commerciaux. Il pourrait être considéré comme étant un maillon de la chaîne de valorisation présentée dans le BeerGame. Cependant, UpStrat apporte beaucoup de détails à la modélisation interne d'une entreprise de production et au lien entre les fonctions stratégiques d'une entreprise. Dans le BeerGame, seule la relation entre entreprises est exploitée.

Les entreprises participantes à une session UpStrat sont toutes concurrentes. Elles doivent se positionner sur un marché commun. Sur le BeerGame, la notion de concurrence est absente et laisse place à la coopération. Le marché est simulé par l'animateur qui tire le marché en déclenchant des commandes au détaillant. Il n'y a donc aucune

répartition d'un marché. Celui-ci est dédié à un élément de la chaîne logistique. Sur UpStrat, la répartition d'un marché global aux différents concurrents est le cœur du système du jeu.

Enfin, l'indicateur de performance du BeerGame est simple et n'est exprimé qu'en coût de stockage et de non-livraison. La composition du score sur UpStrat est multiple et permet à chaque concurrent de sélectionner les indicateurs sur lequel il se concentrera.

1.2 Présentation de la problématique

Le contexte ayant été présenté, nous allons désormais nous focaliser sur l'objectif, les enjeux et les verrous scientifiques à lever pendant l'étude.

1.2.1 Enjeux

Pour permettre un plus large déploiement de ce type de supports pédagogiques, il paraît pertinent de tenter d'apporter une solution aux contraintes évoquées précédemment.

L'objectif de cette étude est de doter le simulateur UpStrat et plus largement les business games, d'agents « intelligents ». Ces agents seront des compléments pédagogiques et des assistants efficaces aux participants et animateurs.

Nous envisageons d'exploiter ces agents dans deux configurations :

- **Conseiller les participants sur des décisions cohérentes** : nous l'appellerons le rôle de « conseiller ». Ce rôle permettra aux équipes de faire appel pour tout ou partie de leurs décisions à l'outil développé. Celui-ci pourrait proposer à l'équipe des décisions optimales en production, distribution, finance. Il pourrait également indiquer un taux de pénétration de marché prévisionnel provenant d'une analyse des décisions historiques des concurrents et des orientations de l'équipe faisant appel à lui. Enfin, il pourrait indiquer aux équipes leur chance de performance au regard de leur décision et ainsi offrir en direct une expertise précieuse à l'équipe et par voie de conséquence à l'animateur. Cette option pourra d'un point de vue pédagogique se révéler très utile pour des publics ne disposant pas des prérequis nécessaires au pilotage complet d'une entreprise.
- **Concurrent des équipes réelles** : nous l'appellerons le rôle de « concurrent ». L'inscription de quelques agents autonomes sur une partie pourrait montrer aux équipes des jeux de décisions cohérentes. Elle permettrait aussi à un individu seul de se « mesurer » à l'intelligence artificielle proposée. Ce rôle de concurrent permettrait aussi de simplifier la formation des formateurs aux règles du jeu. En effet, quoi de plus formateur que de participer à un jeu avant d'en être l'animateur ?

Ces différents usages peuvent être envisagés indépendamment ou conjointement. Nous n'avons proposé que quelques modes d'exploitation de ces agents mais il est évident que la liste pourra être enrichie avec l'usage et en fonction des performances et résultats obtenus.

1.2.2 Les verrous

Les principales difficultés que nous rencontrerons dans cette étude sont :

- la simultanéité,
- un temps de réponse court,

- le principe de modalité,
- l'indicateur de performance.

Nous reprenons dans les quelques lignes suivantes ces différents points afin de les détailler et d'en définir les contours.

Simultanéité

Dans un business game, les décisions des équipes sont prises dans un temps de réflexion banalisé pour toutes les équipes en même temps. La centralisation des décisions se fait à un instant donné simultanément défini en amont de la phase de réflexion. Ce mode de jeu s'éloigne donc des jeux habituels d'échec, de poker... pendant lesquels les concurrents font évoluer chacun leur tour leurs positions. Les décisions des équipes peuvent dans notre cas être assimilées à un ensemble de coups que chaque équipe prépare et joue en même temps sur un plateau de jeu virtuel.

Nous pourrions rapprocher notre étude à certains problèmes issus de la *théorie des jeux*. Nous ferons un rapprochement avec cette thématique de recherche dans le chapitre 4 de ce document.

Nos agents devront donc prendre un ensemble de décisions pour chaque tour et attendre le verdict de la confrontation des décisions de tous les participants pour en évaluer la pertinence et enclencher un nouveau tour.

Un temps de réponse court

Les fonctionnalités que nous proposons de concevoir n'ont de sens que si elles accompagnent les acteurs d'une session : participants et formateurs. Ainsi, les agents seront obligatoirement sollicités pendant une session de jeu. Ils fonctionneront donc sur des périodes bloquées comme des travaux dirigés de 2 heures par exemple et devront proposer leurs conclusions plusieurs fois sur cette période et pour une ou plusieurs équipes. Cette contrainte majeure nous imposera, pendant l'étude, d'intégrer continuellement cette dimension et nous forcera à trouver régulièrement un compromis entre la performance de la méthode, le temps d'exécution et la consommation de ressources allouées.

Un agent qui mettrait plus de temps qu'un participant pour prendre une décision ou proposer une expertise perdrait en crédibilité. Mieux, il serait flatteur que lors de la mise en place de ces outils, les décisions en émanant soient prises avant que les participants n'interviennent.

Dans un rôle de conseiller, il est indispensable que les participants interrogent l'agent avant de prendre leurs décisions.

Des décisions représentées par des modalités

Les choix réalisés dans le simulateur Upstrat sont, pour la plupart, définis sous la forme de modalités. Dans la partie consacrée à la présentation du simulateur, chaque capture d'écran fait état de ces modalités. Elles sont représentées par des entiers.

Par exemple, les prix, pour chaque segment de marché, prennent une valeur comprise dans l'ensemble $\{-4; -3; \dots; +3; +4\}$. La décision en publicité se fait de la même manière. Toutes les décisions opérationnelles, ou décisions financières, sont réalisées de la sorte.

Ce choix de conception du produit a été fait afin que les participants se concentrent sur l'essentiel pendant leur prise de décisions. L'essentiel est, pour ce simulateur, d'être capable de prendre des décisions cohérentes pour atteindre un objectif dans le cadre d'une stratégie donnée. Les concepteurs ont souhaité élargir le champ de réflexion des

participants aux autres dimensions du marketing-mix plutôt que de les laisser se focaliser sur le prix. En effet, cette décision cristallise très souvent l'attention des joueurs aux dépens d'autres dimensions pourtant fondamentales dans un modèle business to business.

Les décisions sont donc toutes des données entières qui imposent des choix. La valeur médiane, et donc réelle, pour une dimension de la décision n'est pas admise. Ce caractère nous conduit donc à la notion de combinatoire de décisions puisque pour chaque modalité sélectionnée, un sous-ensemble de nouvelles décisions induit par le choix précédent verra le jour. Il y a donc de nombreuses combinaisons possibles pour décrire une décision complète.

Le critère de performance

Les jeux d'entreprise ne permettent pas de définir de critère objectif de performance des décisions. Les business games sont des jeux donc il est assez logique qu'il y ait un vainqueur. Cependant, la qualité du vainqueur n'est en aucun cas renseignée par son classement, son nombre de points ou sa performance économique, si l'on sort du contexte du jeu, ces grandeurs. Le vainqueur est désigné parmi ses pairs et de façon relative. Ainsi, il s'agit d'être meilleur que les autres joueurs du même jeu. Être le meilleur ne signifie pas être bon dans l'absolu. Ainsi, il est tout à fait possible, lors d'une partie, que le joueur qui remporte la session soit en situation financière très difficile voire en dépôt de bilan. Son positionnement pourrait être assuré par les piètres performances des concurrents. Inversement, une équipe dont les résultats économiques seraient flatteurs pourrait ne pas être sur le podium d'une partie si tous les autres joueurs sont encore plus performants.

Notons que dans la vie économique réelle, de nombreux scores économiques coexistent. Les états comme les entreprises sont évalués par des agences de notation publiques ou privées. Ces évaluations permettent aux parties prenantes de ces entités de s'assurer de leur solidité actuelle et projetée (créanciers, actionnaires/prêteurs, salariés/citoyens, ...).

Ces scores sont définis à l'issue d'analyses de données macroéconomiques et sectorielles. Ces traitements permettent de sélectionner dans un premier temps les critères à intégrer aux scores. Ils offrent aussi la possibilité de définir les bornes d'acceptabilité de chaque critère et l'importance relative d'un critère par rapport à un autre sur la note finale. Ces scores ne sont pas utilisés pour établir un classement du meilleur état ou de la meilleure entreprise mais plutôt pour évaluer des critères de risque ou d'opportunité bien souvent liés à des risques d'insolvabilité court terme, moyen terme ou long terme.

1.2.3 Les objectifs de la thèse

Les objectifs des travaux de recherche réalisés dans ce manuscrit ont été menés afin de répondre aux verrous préalablement cités. En effet, l'attrait des business games pour le grand public n'est pas encore évident.

L'objectif de la thèse est donc de mettre en œuvre un agent embarqué dans le business game UpStrat.

Cet agent constituera une équipe. Sa mission sera, dans un contexte économique donné, pour un tour de jeu donné, de proposer le meilleur ensemble de décisions possible. Cette proposition devra être formulée rapidement.

Le contexte économique est pour nous l'environnement concurrentiel composé par :

- les autres équipes/joueurs/concurrents,
- les constantes économiques : taxes, impôts, coûts externes,

- les clients de l'entreprise,
- les parties prenantes à l'entreprise : salariés, actionnaires, banquiers, conseils...

Les éléments précédemment cités sont en perpétuelle évolution pendant une partie et imposeront donc à l'agent une capacité d'adaptation permanente.

L'introduction d'agents intelligents au sein des business games est motivée par la nécessité de les faire progresser à l'instar de ce qui peut être fait pour l'industrie du jeu vidéo. En effet, cette industrie crée régulièrement des partenariats avec le monde universitaire afin d'évoluer techniquement et ainsi de survivre face à la concurrence acharnée sur ce secteur en croissance (HAO et collab. [2015]).

Ainsi, les éditeurs de jeux vidéo souhaitent faire progresser leurs modèles, leurs environnements ou encore leurs personnages. Ces évolutions se traduisent par de nécessaires progrès graphiques mais aussi par l'exploitation d'innovations scientifiques. Les concepteurs développent, par exemple, des agents intelligents souvent appelés « intelligence artificielle » (IA) et les intègrent aux produits. Ces agents assurent une meilleure immersion du joueur et augmente le réalisme des applicatifs. Elles doivent régulièrement évoluer et s'adapter d'une part à l'évolution technique et d'autre part aux exigences des joueurs toujours plus affûtés.

La forte demande en jeux de rôle massivement multi-joueurs en ligne a fait apparaître un autre aspect purement économique. En effet, les éditeurs doivent vivre de leurs créations et ainsi monétisent les outils qu'ils développent. Ils essaient de convertir les nouveaux joueurs, exploitant souvent des versions limitées mais gratuites, aux versions complètes et payantes. Ils ont souhaité naturellement essayer de modéliser les comportements des joueurs. Il était important d'identifier les profils susceptibles de passer à l'acte d'achat. Cette opération ne pouvait se faire qu'en offrant l'accès à leurs données de jeu aux spécialistes en traitement de données (DRACHEN et collab. [2014]). Ils ont ainsi tissé des liens étroits avec certains laboratoires de recherche.

1.3 Conclusion

Dans ce chapitre nous avons dans un premier temps présenté le contexte de cette thèse. En particulier, nous nous sommes focalisés sur l'évolution de la pédagogie et l'arrivée massive de méthodes pédagogiques innovantes et alternatives aux amphithéâtres.

Par la suite, nous avons présenté les business games. UpStrat, l'objet de cette étude, a été introduit et comparé à une simulation reconnue scientifiquement : le BeerGame. De plus, nous avons identifié pour les business games quelques difficultés limitant leur exploitation. C'est dans ce contexte que nous avons souhaité proposer des réponses adaptées permettant de fluidifier les sessions de formation et ainsi d'augmenter leurs diffusions.

Cependant, nous avons identifié quelques verrous à lever afin d'atteindre notre objectif. C'est pourquoi, avant de développer une solution performante, nous proposons dans le chapitre 2 notre méthodologie générale.

Nous avons alors observé le fonctionnement d'une équipe de joueurs afin de l'imiter. Par la suite, nous avons adapté ces comportements afin de rendre possible une résolution automatisée du problème combinatoire soulevé.

Chapitre 2

Méthodologie et modélisation

Dans le chapitre précédent, nous avons présenté les business games et les objectifs de la thèse. Nous souhaitons dans ce chapitre présenter notre méthodologie et exposer la modélisation du problème. L'activité opérationnelle majeure d'un business game réside dans la prise de décisions successives. Ainsi, dans ce chapitre nous nous concentrons sur le processus de construction d'un ensemble de décisions que nous nommerons solution.

Pour réaliser correctement cette opération, nous avons analysé les relations et enjeux au sein de l'entreprise et avons proposé une décomposition de la société virtuelle en activités autonomes. Cette décomposition est justifiée par une analyse des différents métiers de l'entreprise que nous avons adossée à une démarche d'analyse stratégique classique. Les grandes étapes de cette démarche sont reprises en début de chapitre. Nous avons également souhaité identifier les dépendances financières qui existent entre les activités des sociétés virtuelles. Nous nous sommes alors ainsi orientés vers la comptabilité de gestion afin de valider nos choix. Nous présentons également les fondamentaux de ce domaine en début de chapitre.

À l'issue de ces rappels théoriques, nous avons souhaité que l'agent autonome et automatisé se comporte comme les différents membres d'une équipe. Nous avons donc étudié les liens existants entre les diverses fonctions opérationnelles de l'entreprise. Nous présentons donc les différentes étapes de la construction d'une décision pour une équipe de joueurs. Ensuite, nous proposons une adaptation de ce cheminement à l'agent décisionnel.

Nous concluons ce chapitre par une présentation de la modélisation déployée sur deux départements opérationnels du business game Upstrat.

2.1 Notions de stratégie

La présentation de notions fondamentales en stratégie d'entreprise nous sera utile pour valider des choix ultérieurs. Nous avons volontairement sélectionné les points clés de ce thème dans un souci de clarté et de cohérence globale du document.

2.1.1 La démarche stratégique classique

Dans le pilotage d'activités économiques, les managers mettent en œuvre une démarche leur permettant d'assurer la pérennité de leur structure. Cette pérennité se traduit naturellement par des objectifs financiers mais aussi en terme de risque et surtout de limitation de risque. Une démarche stratégique est décomposée en plusieurs étapes

(DEMIL et collab. [2013] et JOHNSON [2014]) comme le présente le schéma ci-dessous (figure 2.1).

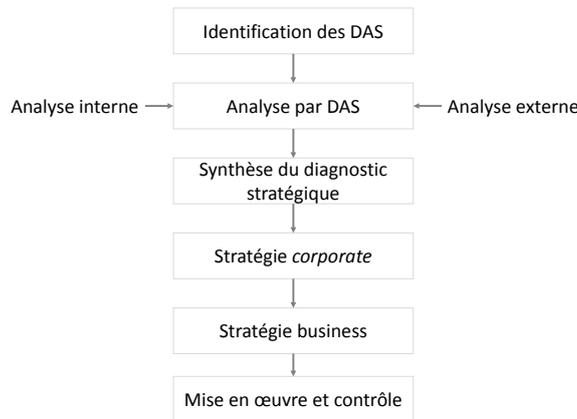


FIGURE 2.1 – La démarche stratégique classique (DEMIL et collab. [2013])

La première étape d’une démarche stratégique réside dans l’identification des *Domaines d’Activité Stratégique* (DAS) de l’entreprise. Le portefeuille de DAS d’une entreprise reprend l’ensemble des activités génératrices de revenus. Nous définirons plus précisément cette notion dans le paragraphe suivant.

La deuxième étape consiste en une analyse interne et externe de l’entreprise. Cette analyse doit être réalisée pour chaque DAS. Elle fournit la capacité stratégique de l’entreprise (forces et faiblesses) et les attentes du marché (menaces et opportunités) pour chaque DAS. Ces éléments sont généralement mis en forme selon une matrice « Menaces Opportunités Forces et faiblesses » (MOFf) comme le présente le tableau 2.1.

TABLEAU 2.1 – Matrice d’analyse MOFf pour un DAS (BRESSY [2014])

	+	-
Interne	Forces	Faiblesses
Externe	Opportunités	Menaces

L’étape suivante permet de synthétiser les diagnostics préalablement réalisés. Cette synthèse alimentera la stratégie *corporate*. Cette stratégie peut aussi se nommer stratégie globale car non différenciée par DAS. La stratégie *business* suit la stratégie *corporate*. Elle est définie par segment stratégique. La dernière étape sera la planification, le déploiement opérationnel des actions et moyens et le contrôle de l’efficacité de ces actions.

Dans ce document, nous ne présenterons que la première partie à savoir la segmentation stratégique en Domaines d’Activités qui nous permettra de justifier nos choix ultérieurs. Cette analyse par DAS est basée sur les Facteurs Clés de Succès que nous présentons dans le paragraphe suivant.

2.1.2 Les Facteurs Clés de Succès

Les facteurs clés de succès (FCS) sont les éléments sur lesquels se fonde, véritablement et en priorité, la concurrence entre les entreprises : de ce fait ces éléments correspondent aux compétences à maîtriser pour être performants en terme de lutte concurrentielle (MEIER [2015]).

L'identification des facteurs clés de succès est une première approche pour effectuer la segmentation stratégique. Aussi, notons que pour chaque DAS, une série de facteurs clés de succès sont à identifier. Ils sont en général différents. Si les FCS sont identiques entre deux DAS, leur importance relative sur le marché segmenté sera nécessairement différente.

De plus, ces FCS évoluent dans le temps. Ils devront en particulier s'adapter à la concurrence, à la demande ou tout simplement au cycle de vie du produit ou du marché. Mc Carthy a proposé de regrouper quelques FCS sous quatre intitulés nommés les « 4P » (MC-CARTHY et PERREAULT [1993]) :

- **Price** : Positionnement prix
- **Product** : Produit, technologie employée ...
- **Promotion** : Communication, publicité
- **Place** : Canal de distribution

Ces 4P sont aussi une définition d'une notion appelée le *mix marketing*. Cette définition du mix marketing évolue régulièrement. Le 7P est venu enrichir la vision du 4P par :

- **Process** : Interaction avec le client
- **People** : Capacité de la force de vente
- **Physical evidence** : Composante matérielle identifiant clairement le produit ou le service

Cette modélisation a été adoptée en 2009 par le Chartered Institute of Marketing, plus importante association mondiale des professionnels du marketing (ETTENSON et collab. [2013]).

D'autres modélisations existent aujourd'hui 4C (Commodity, Cost, Communication, Channel), 7C (Corporation, Commodity, Cost, Communication, Channel, Consommateur, Circumstances) SHIMIZU [2009] ...

Cependant, nous resterons sur le modèle 4P qui est le plus simple et le plus diffusé.

2.1.3 Définir un DAS : Domaine d'Activité Stratégique

Afin de définir une stratégie adéquate et performante, il est primordial que l'entreprise connaisse ses *Domaines d'Activité Stratégique* (DAS). Identifier les DAS d'une activité revient à décomposer celle-ci en un ensemble de sous-activités. Ainsi, un DAS se définit comme un sous-ensemble d'une organisation auquel il est possible d'allouer ou de retirer des ressources de manière autonome. Il correspond à une combinaison spécifique de facteurs clés de succès. Le schéma 2.2 présente une entreprise constituée de trois DAS ayant chacun ses propres objectifs, ressources et facteurs clés de succès. Nous avons ajouté une dimension regroupant des services communs à chacun des DAS dits de *support*. Ces services sont communs à tous les DAS comme du secrétariat, des locaux communs ...

Un DAS est donc un segment de l'environnement de l'entreprise à l'intérieur du secteur dans lequel l'entreprise évolue. Ainsi, l'entreprise peut également être représentée comme un portefeuille de DAS.

Un DAS offre des perspectives de croissance et de rentabilité et nécessite des stratégies d'approche spécifique : l'entreprise va décider d'agir sur tel ou tel DAS en fonction de ses objectifs, de ses décisions et de sa volonté.

On appelle segment stratégique un ensemble de 1 à n lignes de produits partageant les mêmes ressources pour affronter les mêmes concurrents sur les mêmes marchés, avec les mêmes technologies et ayant par conséquent à maîtriser les mêmes facteurs clés de

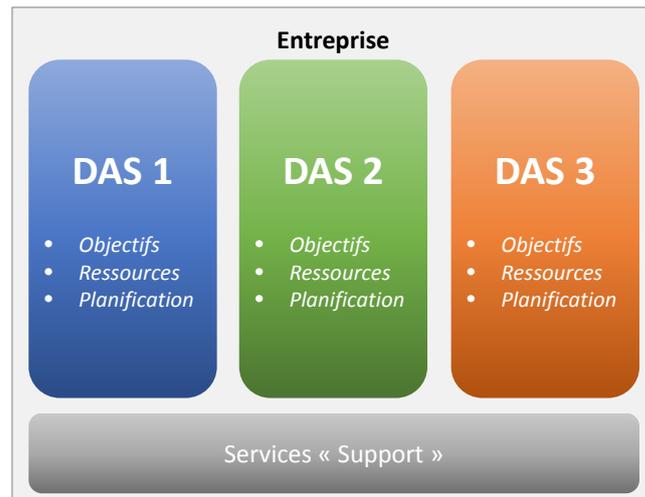


FIGURE 2.2 – Décomposition d’une entreprise en DAS

succès. Le modèle de ABELL (ABELL [1980]), représenté sur la Figure 2.3 permet de mettre en évidence les trois axes permettant de caractériser et positionner les DAS d’une société. Ainsi, le premier axe est dédié au service ou au produit proposé. Le second axe est destiné au marché donc au client du produit ou service. Le dernier axe permet de représenter la technologie ou le canal de distribution. Il permet d’intégrer une dimension assurant la différenciation d’un DAS par rapport à un autre.

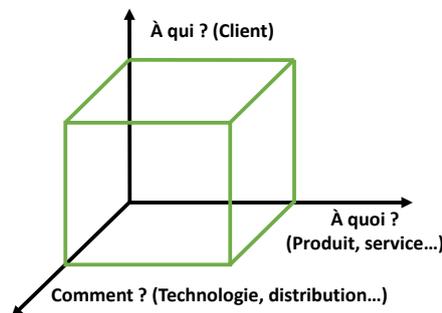


FIGURE 2.3 – Modèle ABELL (ABELL [1980])

Les critères permettant d’identifier des DAS différents se retrouvent souvent parmi les critères suivants :

- Activités homogènes,
- Marchés et type de clientèle,
- Type de distribution,
- Structure des coûts,
- Compétence et technologie requises.

Pour illustrer cette décomposition, nous proposons de reprendre un exemple souvent présenté dans les ouvrages. Ainsi, comme le montre la figure 2.4, un groupe chimique dispose de plusieurs métiers : Pharmacie, Fibres, Colorants et Peinture. Pour chaque métier, nous pouvons identifier divers segments stratégiques : Grand public, Bâtiment, Industrie, Réparation automobile, Construction automobile et Marine.

Nous disposons donc à ce niveau de la décomposition stratégique suffisante pour mener l’analyse par DAS.

Les décompositions présentées en dessous du niveau stratégique permettent d'identifier pour chaque DAS les segments de marché concernés. Il s'agit dans ce cas d'informations marketing que nous n'exploiterons pas dans la suite de ce document.

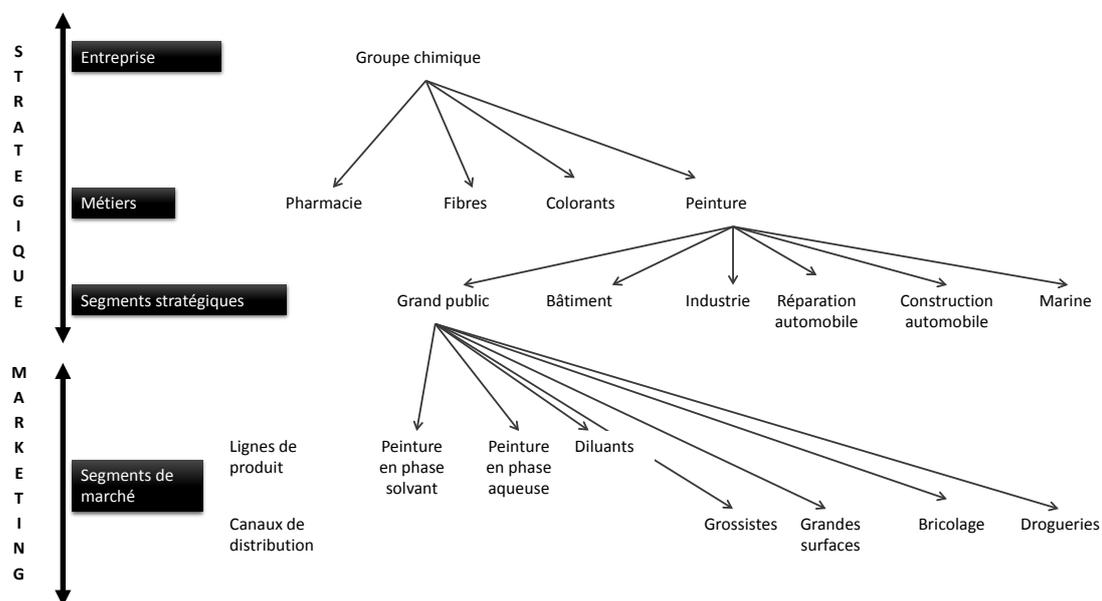


FIGURE 2.4 – Exemple de décomposition stratégique d'un groupe chimique (GARIBALDI [2007])

2.2 Notions de comptabilité de gestion

Dans le paragraphe précédent, nous avons souhaité nous appuyer sur les méthodes classiques d'analyse stratégique afin de décomposer une entreprise en différents Domaines d'Activité Stratégique. Aussi, cette décomposition met en évidence les différents métiers de l'entreprise. Nous souhaitons désormais nous orienter vers une analyse financière des activités. Ainsi, pour réaliser cette opération, nous proposons de nous tourner vers la comptabilité de gestion qui nous offre de très bons outils, que nous rappellerons en début de paragraphe pour l'analyse d'activité d'un point de vue de la « chaîne de valorisation » de l'entreprise.

La comptabilité générale fait un état des flux par nature sur une période de référence, en général un an, correspondant à un exercice comptable. Le regroupement des charges par nature restreint naturellement l'analyse et en particulier empêche la maîtrise de la performance de chaque activité ou service de l'entreprise. La comptabilité de gestion vient pallier les lacunes de la comptabilité générale. Elle permet de rattacher les charges aux activités et donc d'en extraire des informations indispensables à la décision (DUBRULLE et JOURDAIN [2013], PIGET [2015] et SAADA [2005]).

2.2.1 Définitions et nomenclature

Il est nécessaire de disposer du vocabulaire de la comptabilité de gestion afin d'identifier la typologie des coûts.

Un **coût direct** est un coût dont l'origine est un produit ou un service de l'entreprise (par exemple : consommation de matières premières, commission commerciale...). Un **coût indirect** est, quant à lui, généré par le simple fait que l'entreprise soit en activité.

Citons pas exemple le cas des abonnements à l'énergie, la location de locaux professionnels...

Concernant UpStrat, le Tableau 2.2 reprend quelques coûts classés en coûts directs et indirects. Nous avons également choisi de créer différents **centres d'analyses** : Production, Support et Distribution afin d'être en mesure d'appliquer pour chacun d'eux une ventilation des charges indirectes spécifiques.

TABLEAU 2.2 – Typologie des coûts dans UpStrat

Centre d'analyse	Coût direct	Coût indirect
Production	Salaires du personnel (CDI, CDD ...) Achat de matières premières Dotation des lignes de production	
Distribution	Publicité Transport Finition Frais commerciaux	Dotation des succursales
Support	Formation Développement Dotation du développement	Frais de gérance Salaire des administratifs Dotation des constructions

2.2.2 Méthode des coûts complets

Afin d'évaluer le bénéfice réalisé par la vente d'un produit, il faut identifier et répartir les coûts sur chaque produit. La répartition des charges directes ne présente pas de difficultés, car elles peuvent être imputées directement aux produits de façon unitaire. Les coûts indirects, par définition, ne sont pas facilement ni spontanément *affectables* à un produit. Il faut élaborer un outil permettant de les répartir équitablement sur chaque activité de l'entreprise. Cette opération se nomme l'imputation.

L'imputation consiste à répartir ces coûts sur plusieurs postes, selon des clés de répartition. Ces clés sont appelées Unité d'Œuvre (UO). Elles sont fixées plus ou moins arbitrairement par l'entreprise. Il n'existe pas d'unité d'œuvre pouvant être utilisée dans tous les cas. Les UO dépendent de l'activité, de la stratégie, de la volonté du gestionnaire... L'heure de main-d'œuvre directe est une des clés les plus fréquemment utilisées pour répartir les charges indirectes des ateliers de production. Le chiffre d'affaires généré par un produit est une autre clé de répartition pertinente pour les charges indirectes de distribution.

Une fois toutes les imputations des coûts indirects réalisées, il est envisageable de parler de *coûts complets*. Ce coût complet regroupera l'ensemble des coûts directs et indirects imputés à un produit.

Dans la plupart des cas, les contrôleurs de gestion, en charge de la comptabilité de gestion, divisent l'entreprise en centres d'analyse. Ces centres d'analyse représentent des groupements de coûts identifiables dans une chaîne de valorisation d'un produit ou service. Ainsi, l'on retrouve régulièrement un centre d'analyse en production, un autre en distribution qui reprend les charges commerciales et de distribution. Il est parfois opportun d'ajouter d'autres centres d'analyse regroupant par exemple l'ensemble des activités d'amélioration continue, de recherche, de qualité. Le centre d'analyse « support » reprend ces différents coûts.

Ainsi, pour chaque centre d'analyse, il est aisé d'identifier pour chaque produit ou service son coût de production, de distribution ou de support unitaire. La somme de ces trois indicateurs offre le coût de revient du produit ou service.

Notons qu'il existe de nombreuses méthodes de calcul de coûts et en particulier de coûts complets (UVA, ABC ...) (DUBRULLE et JOURDAIN [2013] et PIGET [2015]). Notre objectif ici n'est pas de calculer les marges mais avant tout d'être capable de dissocier les coûts en centres d'analyse afin d'évaluer l'indépendance des produits par rapport à l'ensemble de la structure.

2.2.3 La division de l'activité en produits distincts

Nous souhaitons évaluer la pertinence de la décomposition de l'activité de l'entreprise simulée par UpStrat en références de produit. Pour en évaluer le risque et éventuellement l'opportunité, nous proposons de suivre la démarche de classification des coûts comme présentée précédemment et d'y ajouter un contrôle de représentativité des coûts indirects par rapport à l'ensemble des coûts de l'entreprise.

La figure 2.5 reprend la méthodologie que nous proposons permettant d'identifier si la décomposition est justifiée ou non.

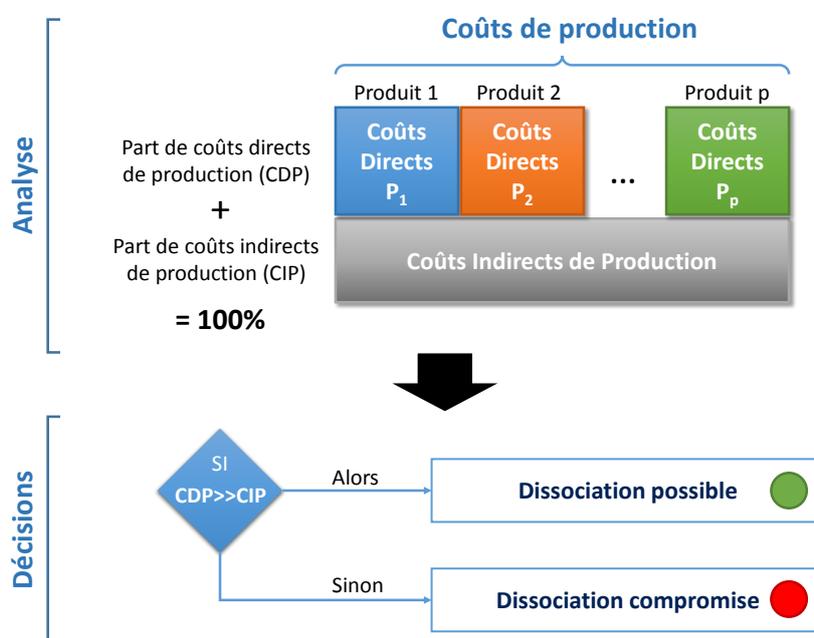


FIGURE 2.5 – Arbitrage sur la décomposition de l'activité

Nous proposons donc dans un premier temps de dresser un état des lieux des différents coûts directs (par produit) et indirects pour la structure (éventuellement par centre d'analyse). Nous proposons d'évaluer la part que représentent les coûts directs par rapport aux coûts indirects pour l'entreprise.

Si la proportion de coûts indirects est très importante par rapport aux coûts directs, la décomposition des activités sera impossible. Si cette décomposition s'avère impossible, les traitements et optimisations devront intégrer tous les produits de l'entreprise lors d'un traitement unique. Dans le cas contraire, une décomposition sera conseillée.

Afin d'illustrer la méthode d'analyse de coûts sur le simulateur UpStrat, nous avons observé la structure financière d'une entreprise présentant des caractéristiques classiques sur un scénario de marché à trois produits. Ces données ont été extraites d'une situation d'entreprise réalisée par une équipe au tour 5. Nous avons choisi ce tour de jeu et cette entreprise car ils intègrent toutes les possibilités offertes par ce scénario à savoir une implantation sur les trois produits et les trois marchés. Ainsi, nous disposons d'un tableau

TABLEAU 2.3 – Exemple de tableau des coûts UpStrat pour une équipe en cours de jeu

	Produit 1	Produit 2	Produit 3	Total
Coûts affectés en production	1 518	2 034	4 069	7 621
Coûts directs				
Achats (Matières / Produits)	970	1 142	2 185	4 297
Personnel	398	629	1 404	2 431
Dotation (lignes)	150	263	480	893
Coûts affectés en administration	540	1 233	2 122	3 895
Coûts directs				
Développement	100	500	500	1 100
Dotation (développement)	52	107	162	321
Formation	0	0	0	0
Coût indirects				
Salaires adm.	299	482	1 126	1 907
Prime	19	31	72	122
Formation	27	44	102	173
Gérance	41	66	154	261
Dotations (constructions)	2	3	6	11
Coûts affectés en distribution	1 488	2 126	4 477	8 091
Coûts directs				
Publicité	378	852	1 991	3 221
Transport	174	198	453	825
Finition	169	250	980	1 399
Frais commerciaux	747	798	1 001	2 545
Coût indirects				
Dotation (succursales)	20	28	52	100

« réaliste ». Le tableau 2.3 identifie les coûts directs et indirects pour chaque centre d'analyse. Les coûts indirects de production et de support ont été imputés selon la clé proportionnelle au chiffre d'affaires généré par le produit.

TABLEAU 2.4 – Répartition des coûts sur UpStrat pour un exemple de jeu

	k€	%
Coûts indirects	2 574	13%
Coûts directs	17 033	87%
Total des coûts	19 607	100%

Nous remarquons que le centre d'analyse d'administration recueille la quasi totalité des coûts indirects. Nous avons, dans le tableau 2.4, calculé globalement, pour l'ensemble de l'entreprise, la proportion de chaque type de coût par rapport à l'ensemble des coûts. Nous observons que les coûts indirects représentent 13% des coûts de l'entreprise.

Nous avons également souhaité effectuer cette analyse sur d'autres jeux afin d'obtenir une vision globale. Cette classification a été menée pour 50 équipes sur 5 tours de jeu (pour les tours 1 à 5). En moyenne, sur ces 50 essais, la proportion de coûts indirects est comprise entre 8% et 14%. La borne inférieure de 8% correspond au début de simulation. Elle peut aussi se retrouver pour les équipes qui préfèrent la sous-traitance à la production. En effet, la sous-traitance est un coût direct et en début de jeu, les équipes n'ont pas d'autre choix que d'y recourir.

Ces résultats nous permettent d'autoriser la décomposition de l'entreprise en produits. Nous sommes conscients que nous intégrons un biais via cette décision. Cependant, la contrepartie de ce biais est de pouvoir décomposer le problème et ainsi proposer de meilleures méthodes pour chacun d'eux. Nous privilégierons le temps d'optimisation à l'optimalité de la solution offerte. Nous pourrions donc traiter indépendamment et éventuellement en parallèle les produits lors des phases d'optimisation.

2.3 La décision du joueur

Après avoir présenté des éléments théoriques de base nous permettant de valider nos choix ultérieurs, nous proposons de nous focaliser sur le fonctionnement d'une équipe de joueurs. Une équipe de joueurs est un ensemble d'individus. Chaque individu effectue une mission sur l'activité ou sur un produit ou service. Ainsi, nous identifions les différents rôles fonctionnels au sein d'une entreprise. Nous présentons également une décomposition des rôles ainsi qu'une analyse de leurs interactions. Cette opération est indispensable car elle nous permettra d'adapter les agents décisionnels au fonctionnement de joueurs réels.

2.3.1 Répartition des rôles

Dans un premier temps, définissons les termes de rôle et de département. Une équipe dans un jeu est en général composée de plusieurs membres d'un comité de direction. Chaque membre de ce comité occupe un rôle au sein du comité de direction mais aussi un rôle opérationnel. On retrouve classiquement un directeur de production, un responsable commercial et marketing, un directeur financier, un responsable R&D ou progrès, un directeur des ressources humaines et enfin un directeur général. Ces rôles sont tous exercés au sein de départements : département Production, département Finance, département R&D . . .

Chaque participant endosse ainsi un rôle opérationnel propre et identifié dans son équipe. Il deviendra le garant du bon fonctionnement de son département. Il maîtrisera en particulier parfaitement les règles du jeu et les indicateurs de son département. Il entretiendra des relations avec les autres membres du comité de direction et avec la direction. Sa capacité de communication avec les autres membres de l'équipe sera un atout majeur permettant d'assurer la performance de l'entreprise.

Le directeur général doit, quant à lui, être le chef de projet. Il vérifiera la conformité des décisions par rapport à la stratégie choisie par l'équipe (voir figure 2.7). Il sera le « gouvernail » de l'équipe et le régulateur des décisions. Son rôle d'arbitre interne imposera rigueur et vision d'ensemble. Il aura aussi un rôle en dehors de l'équipe. Il observera et anticipera le marché et le positionnement des concurrents. Ces fonctions le positionneront à la frontière de l'entreprise. Il surveillera également des indicateurs par fonction (balanced scorecard) (COOPER et collab. [2016]).

Nous citons dans le tableau 2.5 quelques ratios de pilotage et les classons par fonction opérationnelle. Cette liste n'est pas exhaustive et dépend de l'objectif de l'entreprise et donc de sa stratégie. Cependant elle est suffisamment large pour mettre en évidence les points de vigilance d'une entreprise (industrielle dans notre cas).

Le directeur général définira la politique prudentielle en matière de finance, de gestion des stocks. . . Enfin, il mettra à jour les tableaux de bord prospectifs et analysera les écarts entre les prévisions et la réalisation. Cette analyse des écarts pourra se faire selon la méthode *PRMV* qui se focalise sur les variations des Prix, des Rendements de production, des Compositions (pour le M de Matière) et des Volumes produits ou vendus.

2.3.2 L'entreprise, un ensemble de processus en interaction

Nous avons choisi pour notre approche d'optimisation de décomposer le processus de prise de décisions en différents rôles clés. Chaque rôle correspond à une fonction d'un membre de l'équipe. Ainsi, nous divisons le problème complet en différents sous-

TABEAU 2.5 – Ratios de pilotage par fonction opérationnelle

Fonction	Indicateur / Ratio
Production	Productivité
	Vétusté de l'outil de production
	Taux d'activité de l'outil de production
	Niveau de stocks
Commercial	Part de marché
	Marge commerciale
	Volume prévisionnel
	Taux de livraison
Finance	Endettement
	Liquidité
	Coût de revient
	Compte client
RH	Climat social
	Rémunération
	Qualification

problèmes que nous optimiserons séparément. Nous porterons un soin particulier aux relations entre ces éléments. Cette analyse des relations permettra de mettre en œuvre la notion de processus que nous avons représentée sur la figure 2.6. Ainsi, chaque sous problème représentera un processus qui se caractérisera par des données d'entrée, une optimisation et des données de sorties. Ces données de sorties seront les données d'entrée du processus suivant.

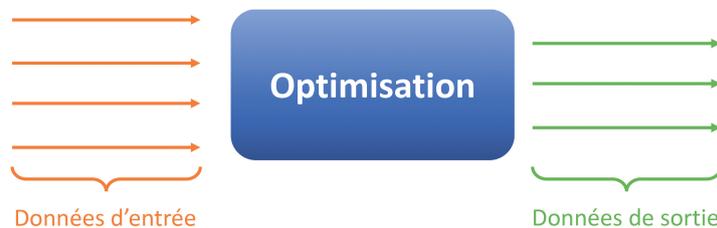


FIGURE 2.6 – Représentation d'un processus

Ce partitionnement nous permettra, dans le rôle de coach, de proposer aux participants des aides adaptées à certains départements sans pour autant intervenir sur tous les départements de l'entreprise virtuelle et donc sur la décision globale de l'équipe.

Lors de l'exploration du problème initial, nous pensions qu'il était opportun de considérer l'entreprise comme une entité unique à traiter dans sa globalité. Cette première approche imposait une optimisation unique et simultanée de toutes les décisions de tous les départements. Cependant, après l'étude approfondie des règles du jeu du simulateur et la confrontation du modèle aux théories de stratégie d'entreprise et de comptabilité de gestion, nous avons validé qu'un **partitionnement** de l'optimisation générale en sous ensembles **indépendants** à optimiser eux-mêmes était recevable et préférable.

Le **partitionnement** présenté va nous permettre d'accélérer les traitements. Il nous imposera cependant de veiller aux relations entre les sous-ensembles et à la cohérence ou faisabilité globale de la solution. Ce partitionnement a également comme avantage de calquer le mode de jeu des équipes.

2.3.3 Relations au sein de l'entreprise

Chaque département issu de la décomposition est un processus. Il se caractérise par des données d'entrée. Leur traitement au cœur du département permet d'obtenir les données de sorties. Il communique avec les autres départements de l'entreprise et y entretient une relation de client ou de fournisseur.

Les départements mis en évidence sur la figure 2.7 sont :

- **Production** : achat/cession de ligne de production, planification de la production, définition du type de ligne de production,...
- **Commercial / livraison** : dimensionnement de la capacité de livraison, type de points de vente,...
- **Finance** : emprunts, dividendes, augmentation de capital,...
- **Recherche et Développement** : investissements en recherche

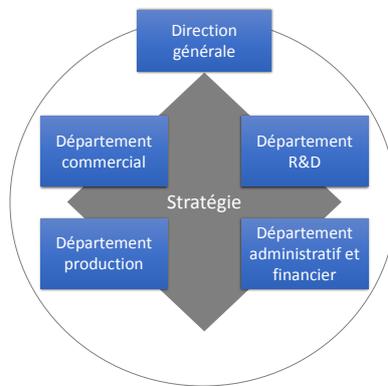


FIGURE 2.7 – Modélisation d'une entreprise en départements principaux

2.3.4 Le joueur et sa solution

La construction d'une solution

Avant d'initier le processus de construction et de sélection des solutions, nous pensons qu'il est nécessaire de nous pencher sur la méthodologie qu'un participant/joueur adopte ou est censé adopter. Nous proposerons ensuite de faire le parallèle avec les étapes d'élaboration que notre agent suivra.

Définissons dans un premier temps ce que nous appellerons *solution*.

Une solution est constituée d'un ensemble de décisions concurrentielles et opérationnelles nécessaires au bon fonctionnement de l'entreprise pour un tour donné ou pour un ensemble de tours donnés.

Cet ensemble comprend entre autre le choix des prix des produits, la publicité, les options de différenciation, les investissements en R&D et les délais de paiement. Ces décisions sont différenciées pour l'ensemble des couples produit/marché étudiés. Nous les regroupons dans le Tableau 2.6.

De plus, il est indispensable d'intégrer une dimension temporelle dans les solutions créées. En effet, un joueur doit prendre des décisions pour le cycle actuel mais il doit aussi être capable d'anticiper les cycles suivants. Plus l'anticipation est importante, plus la stratégie planifiée a des chances de succès. Cette anticipation permettra également de prendre les décisions les plus performantes sur un plan comptable.

TABLEAU 2.6 – Description d’une solution

$price_{p,z,n}$	Prix de p sur le marché z au tour n
$advertising_{p,z,n}$	Publicité pour p sur le marché z au tour n
$option_{p,z,n}$	Options pour p sur le marché z au tour n
$rd_{p,n}$	Efforts en R&D pour p au tour n
$payment_term_{z,n}$	Délai de paiement pour z au tour n

Ainsi, nous avons schématisé sur la Figure 2.8 les différentes étapes de construction d’une solution.

Sur le premier niveau (Étape 1), les décisions du tableau 2.6 décomposées pour les tours n à $n + p$ sont prises. Nous nommons cet ensemble de valeurs : *hypothèse*. En effet, à ce stade de la génération d’une solution, nous ne pouvons évaluer la pertinence de cet ensemble de valeurs.

En fin d’étape 1 du processus de construction, la solution n’est qu’une série d’hypothèses et de choix commerciaux/marketing sans lien avec le marché. Cependant, elle comporte des choix pour tous les couples produit/marché et pour l’ensemble des tours à anticiper.

L’étape 2 de la construction d’une solution (Figure 2.8) est consacrée à l’anticipation de la réaction des marchés ou des concurrents au regard des décisions prises à l’étape 1. Cette étape produira des commandes prévisionnelles en comparant les décisions générées à l’étape 1 aux décisions des concurrents. Nous disposerons donc, en fin d’étape 2, pour chaque cycle anticipé et par couple produit/marché, d’une valeur entière offrant les commandes prévisionnelles en lien avec les positionnements commerciaux définis en étape 1.

Ces commandes seront les livrables de l’étape 3 de cette même figure. Les départements Production, Distribution et Finance s’adapteront aux volumes prévisionnels des commandes. Chaque département y dimensionne son activité afin d’honorer les commandes prévisionnelles issues de l’étape 2 tout en ayant pour objectif de réduire le coût global des opérations. Ces optimisations, formalisées ou non, ajoutent une nouvelle fois un ensemble de décisions pour chaque tour à la solution. Précisons que le département Finance n’est pas contraint à une prévision au delà du tour courant. Il devra régulièrement s’adapter aux résultats obtenus du cycle précédent.

A l’issue de l’étape 3, nous obtenons un ensemble de décisions cohérent sans information sur sa qualité concurrentielle. Cet ensemble prend donc le nom de *solution*.

Précisions sur la construction d’une solution

Après avoir présenté les principales étapes menant à une solution, nous souhaitons apporter quelques explications complémentaires sur les éléments prépondérants de ce processus.

Pendant la première phase d’élaboration d’une solution, notons que des choix réfléchis pour les tours $n + p$ avec un p élevé optimiseront les moyens mis en œuvre. Cependant, plus le nombre de cycles prospectifs intégrés sera élevé plus il sera complexe de construire cette solution. Cette complexité sera traduite par un temps d’optimisation élevé provoqué par une explosion combinatoire.

En pratique, il est difficile pour un participant de proposer une planification supérieure à 2 cycles. Bien souvent, les participants analysent l’historique du dernier tour et

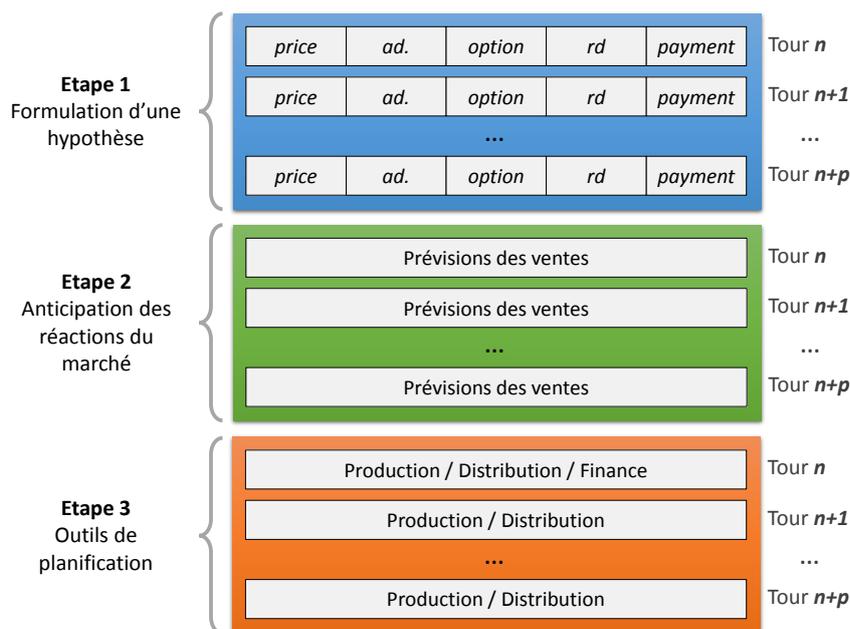


FIGURE 2.8 – Construction d'une solution par un joueur

planifient pour le cycle de jeu en cours. Cette observation se vérifie généralement pour les premiers tours de jeu. L'anticipation des joueurs apparaît avec l'expérience alimentée par les succès et les échecs des équipes.

Il est important de souligner que l'étape 2 est fondamentale. Si les prévisions des ventes sont trop importantes, elles entraîneront un surdimensionnement des moyens de production et de distribution lors de l'étape 3. Elles déboucheront naturellement sur des problèmes de stocks importants et donc de surfinancement du besoin en fonds de roulement par le département Finance.

L'étape 2 impose réflexions et analyses croisées toujours fastidieuses et incertaines dans l'environnement économique simulé. L'expertise exigée sur l'étape 2 sera consolidée par ses connaissances en marketing, finance d'entreprise... L'équipe pourra aussi s'appuyer sur son expérience du jeu. Elle lui donnera des informations sur les réactions du modèle concurrentiel intégré au simulateur. Enfin, les équipes s'appuieront aussi sur des études de marché qui présenteront les positionnements des concurrents. Cette source d'informations offrira à l'équipe des clés de lecture concernant la stratégie des concurrents et la pertinence de ses choix passés. Notons qu'en début de simulation, les participants sont sans information et optent en général pour des décisions médianes minimisant les risques et assurant une survie à court terme.

Enfin, l'étape 3 constitue un moment fondamental dans la construction d'une solution. En effet, lors de la planification, il est possible que les décisions produites pendant les étapes 1 et 2 ne soient pas réalisables par manque de financement, par sous capacité de production ou encore par une capacité de distribution trop faible. Cette étape pourra remettre en cause la faisabilité complète de la solution.

Le choix d'une solution

Dans le paragraphe précédent, nous avons présenté un processus normal de construction d'une décision. Ce processus est souvent répété plusieurs fois par les équipes à partir de formulations d'hypothèses différentes. Il entraîne donc nécessairement la cons-

truction d'un ensemble de solutions.

Ces hypothèses initiales différentes sont générées par la modification d'un seul élément/décision d'une hypothèse précédente. Dans ce cas, il s'agira d'une solution finale très proche de la solution source mais qui permettra de s'ajuster au résultat final attendu. Dans d'autres cas, les hypothèses peuvent être totalement différenciées et proposer par exemple le lancement d'un nouveau produit ou la conquête d'un nouveau marché. D'autres fois encore, les équipes imaginent des scénarios optimistes ou pessimistes.

Ces différentes orientations, une fois consolidées par les décisions de production, de distribution et financières, constituent de nouvelles solutions cohérentes et réalisables. Une seule solution sera ensuite choisie par l'équipe et confrontée aux solutions proposées par les concurrents pour le tour courant du jeu comme présenté sur la figure 2.9.

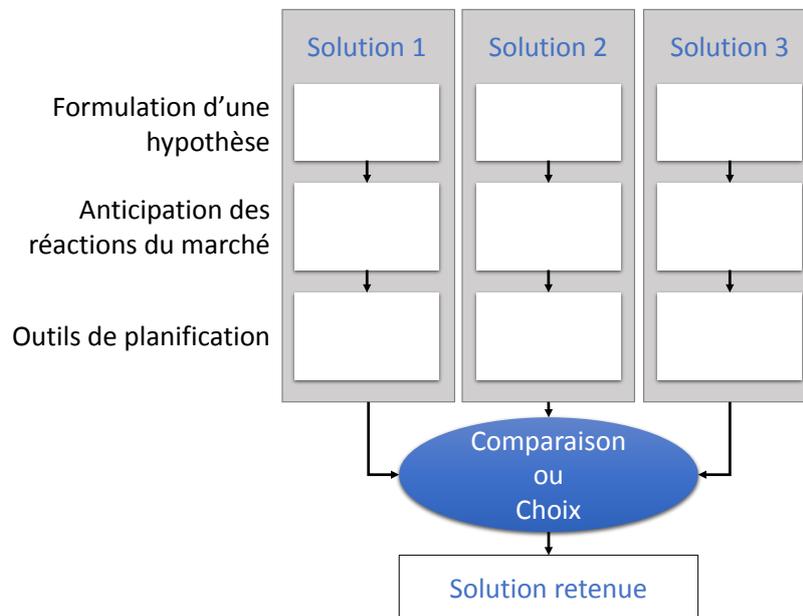


FIGURE 2.9 – Modèle décisionnel d'un comité de direction

2.4 Méthodologie adoptée par l'agent

Nous souhaitons que les agents s'approchent le plus possible d'un joueur réel pour construire une solution. Ainsi, dans ce chapitre, nous présentons les étapes proposées pour la construction automatique d'une solution. Nous présentons ensuite une approche de résolution par une méthode d'optimisation hybride.

2.4.1 La construction d'une solution complète

Comme présenté dans le paragraphe précédent, les solutions des joueurs sont construites successivement au cours de plusieurs étapes. L'approche représentée par la figure 2.10 reprend ces différentes étapes tout en les adaptant au format d'un agent autonome. On y distingue trois « macro étapes » :

1. Formulation d'une hypothèse
2. Optimisation des départements Production et Distribution
3. Optimisation du département Finance

Entre chacune de ces étapes, des computations sans optimisations sont lancées. Elles permettent en particulier de simuler la loi de marché et ensuite de générer les états financiers de l'entreprise si l'on y intègre les décisions en cours de génération.

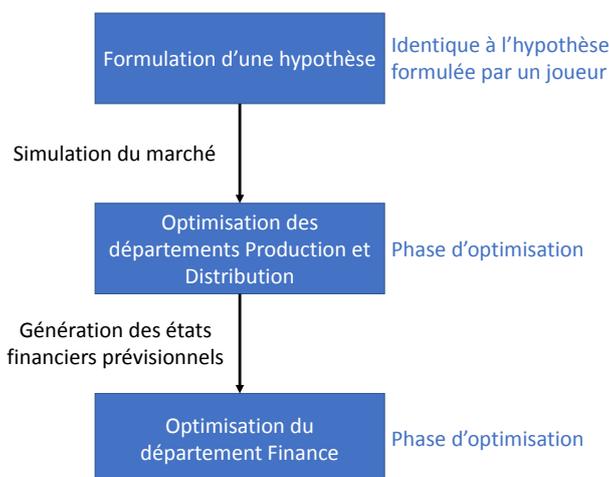


FIGURE 2.10 – Construction d'une solution par un agent

La Figure 2.11 reprend en détail ces différentes étapes et présente les outils d'optimisation que nous avons souhaité adjoindre à chaque étape.

La première étape d'optimisation est dédiée à la génération d'une hypothèse. Elle est réalisée au moyen d'une recherche locale.

La simulation de la réaction du marché est l'étape suivante. Celle-ci est une étape de computation. L'objectif de cette computation est, à partir d'une hypothèse générée, de produire des commandes prévisionnelles. Ces commandes proviendront de la mise en concurrence et donc de la loi de marché intégrée au simulateur.

L'optimisation suivante, optimisée au moyen de programmes linéaires mixtes, permettra de résoudre les départements Production et Distribution. Cette étape donnera l'ensemble des décisions d'activité de l'entreprise. Seules les décisions financières seront absentes des solutions à ce stade de la construction.

L'étape de computation suivante permettra donc de réaliser les états financiers prévisionnels qui permettront de lancer l'optimisation du département Finance. Cette optimisation est également réalisée au moyen d'un programme linéaire mixte.

Une fois les décisions financières rendues, la génération des états financiers définitifs est possible et conduit à la solution finale.

Les étapes de computation correspondent à des calculs internes à l'entreprise comme le calcul de coûts, l'établissement des tableaux de bord financiers : bilan, compte de résultat et structure de financement. Ces étapes, bien que parfois lourdes, n'imposent aucune optimisation et ne présentent aucune difficulté. Elles demandent d'appliquer les règles comptables et de contrôle de gestion au modèle d'entreprise choisi.

Dans le contexte imposé par les hypothèses, les étapes d'optimisation sont nécessaires pour déterminer les décisions les moins coûteuses pour l'entreprise. Le résultat attendu après ces processus d'optimisation sera donc un ensemble de décisions qui viendront enrichir la solution. Cet ensemble, issu d'une hypothèse à l'origine de la solution, répondra aux contraintes imposées par l'hypothèse et par les règles du jeu.

Si pendant la construction de la solution, une des phases d'optimisation échoue, nous serons en présence d'une solution non réalisable. L'hypothèse de départ sera alors rejetée

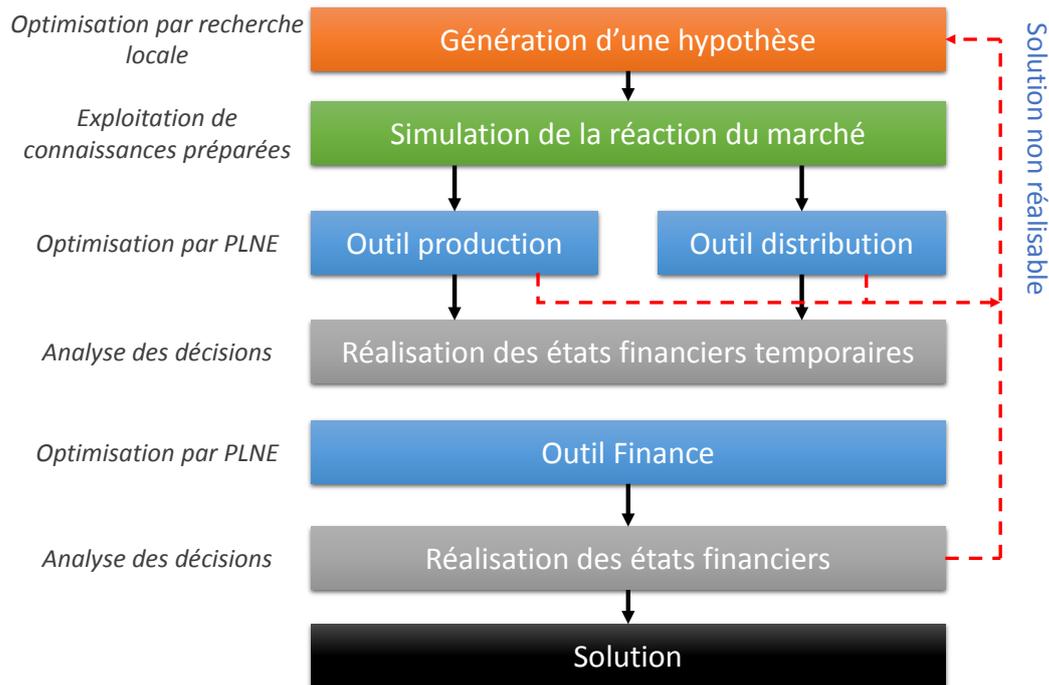


FIGURE 2.11 – Génération d'une solution

dans sa totalité et la construction de la solution sera immédiatement stoppée. L'échec de l'optimisation peut provenir de plusieurs causes :

- **Temps de calcul trop élevé** : La résolution de l'optimisation est interrompue après un temps de calcul fixé arbitrairement puisque la construction d'une solution demande plusieurs optimisations et que la réponse doit être donnée en un temps court,
- **Contrainte(s) non satisfaite(s)** : L'optimisation ne peut aboutir car une ou plusieurs contraintes sont non satisfaites, c'est-à-dire, qu'il n'existe pas de solution réalisable étant données les hypothèses formulées au départ

Afin de réduire le temps de construction d'une solution, nous avons parallélisé l'optimisation des départements Production et Distribution. Nous avons considéré que ces départements étaient indépendants l'un de l'autre. Dans les faits, tous les produits (pour la production) et les marchés (pour la distribution) sont exécutés dans des processus différents.

Le lancement de l'optimisation par PLNE pour « résoudre » le département Finance doit attendre la fin de la résolution des départements Production et Distribution avant de profiter de leurs résultats.

Dans ce contexte, l'objectif global de notre méthode réside dans la découverte de la meilleure solution (qui correspond aux décisions d'une équipe pour un ensemble de tours donnés). A ce stade, nous ne disposons d'aucun gage de qualité concernant la solution. Nous ne disposons que d'un ensemble de solutions qui correspondent aux règles du jeu et qui optimisent les coûts des différents départements de l'entreprise. Il n'y a pas de fonction objectif globale.

Afin d'obtenir des solutions performantes, nous devons tester de nombreuses hypothèses. L'ensemble des hypothèses, appelé *espace de recherche*, sera détaillé dans le cha-

pitre suivant. Nous devons également établir une stratégie de sélection des hypothèses au sein de l'espace de recherche et ainsi définir un moyen de sélection des meilleures. Nous proposons de déployer une méthode de recherche locale pour remplir les missions de parcours et de sélection des hypothèses.

La présentation de la méthode de recherche locale proposée fera l'objet du chapitre suivant. Cependant le prochain paragraphe en donne les grandes lignes.

2.4.2 Proposition d'une approche de résolution par méthode hybride

Dans cette étude, nous nous heurtons à une principale contrainte qui est la rapidité de l'optimisation. Elle doit être plus rapide que la décision d'une équipe. Nous ne sommes cependant pas dans une obligation de calcul en « temps réel ». Un temps de résolution raisonnable se traduit par une optimisation inférieure à deux minutes. Cette barrière est proposée au regard de l'exploitation de l'optimisation en mode *coaching* (accompagnement des équipes). En effet, les participants ne peuvent attendre trop longtemps une réponse à leur appel à l'aide. Dans le mode *concurrent*, les participants réels ne peuvent prendre des décisions éclairées en moins de 20 minutes. Ainsi, le processus n'est pas contraint. Cependant, nous appliquerons la contrainte temporelle la plus forte à la résolution du problème complet soit 120 secondes au maximum.

De plus, il n'est pas nécessaire que notre solution soit la meilleure. Nous ne sommes pas à la recherche de l'optimum global. Cependant une très bonne solution est attendue. C'est pourquoi, notre approche globale sera heuristique. Elle permettra de combiner une solution en temps raisonnable, tout en intégrant pour quelques sous-problèmes une résolution exacte. Ainsi nous proposons le développement d'une méthode hybride, basée sur un schéma de recherche locale intégrant une résolution exacte (à travers la résolution de programmes linéaires mixtes) pour certains sous-problèmes.

Pour initier la démarche et la procédure de recherche de solution, la recherche locale démarre par une solution initiale (S_0). Nous pouvons prendre, par exemple, la moyenne observée des décisions de joueurs réels sur des parties déjà jouées pour le premier tour de jeu. La solution S_0 représente donc une hypothèse crédible. Sa faisabilité sera évaluée de prime abord après être passée par toutes les étapes de construction et de validation définies dans le paragraphe précédent.

Une fois cette première solution identifiée, p autres hypothèses dans le voisinage de la solution initiale sont générées. Toutes les hypothèses suivront les étapes décrites dans la génération d'une solution complète (Figure 2.11). A l'issue de cette étape, p solutions réalisables (ou moins) sont obtenues. Elles seront ensuite comparées au cours d'un tournoi interne. Rappelons qu'il n'est pas possible de définir de critère absolu de performance permettant d'évaluer la qualité d'une solution. Cependant, nous basant sur les règles du jeu, il est possible de comparer, dans un contexte donné, plusieurs solutions entre elles et de déterminer la meilleure.

Les solutions seront toutes comparées à l'issue de leur participation à un tournoi. L'objectif de ce tournoi sera d'établir un classement des solutions générées. Ce tournoi exploitera les règles de classement présentées dans le premier chapitre par la figure 1.18. Nous serons ainsi capables de dire qu'une solution sera meilleure qu'une autre dans le contexte proposé. Le contexte appliqué à la sélection de la meilleure solution est identique au contexte de détermination de la meilleure équipe sur un tour de jeu.

Ainsi, nous obtenons la meilleure des solutions parmi les solutions générées. Cette solution sera donc conservée et sera considérée comme étant la solution courante pour l'itération suivante et donc pour construire de nouveaux voisins. Ce mécanisme donne

naissance à une méthode itérative.

Dans cette approche une solution correspond à ce que nous avons appelé « hypothèse » précédemment. Sa représentation doit donc décrire les décisions à prendre pour un ensemble de tours de jeu et pour tous les couples produit/marché. Par conséquent, sa représentation proposée consiste en un vecteur d'entiers reprenant entre autres le prix, le niveau de la publicité, les efforts de R&D ... de chaque produit, sur chaque marché pour les n tours anticipés.

Cet article a fait l'objet d'une communication scientifique (DUFOURNY et DHAENENS [2015]).

2.4.3 Le tournoi comme outil d'évaluation

Comme explicité précédemment, la performance d'une équipe ne peut être considérée qu'au regard des autres participants du même univers économique/concurrentiel. Ainsi, il n'est pas possible de comparer deux équipes provenant de deux jeux différents. Nous ne disposons pas, contrairement à de nombreux problèmes d'optimisation de la littérature, de critère de performance absolu. Dans le chapitre précédent, un paragraphe a été consacré à l'établissement du critère de scoring.

Ce score ne montre pas la véritable performance économique des équipes. Il est courant que toutes les équipes d'une session affichent des pertes d'exploitation et obtiennent un score identique à d'autres équipes sur d'autres sessions rentables et présentant des bénéfices.

Les scores des vainqueurs des jeux ne sont pas non plus comparables les uns par rapport aux autres. En effet, en cas de leadership partagé sur une session entre différentes équipes sur plusieurs tours, le score global obtenu par le vainqueur sera réduit. Les écarts avec les poursuivants le seront aussi.

Enfin, le score moyen évolue aussi en fonction du nombre de concurrents en lice. Dans un groupe comptant un effectif réduit, l'équipe la plus faible empochera quand même quelques points. Dans une session très chargée, le dernier du jeu enregistre très souvent un score global très faible.

Dans une configuration de formation, le score permet surtout de motiver les participants. Il sollicite leur combativité et leur permet de mettre en lumière des ratios et indicateurs importants pour le pilotage d'une activité.

Puisque nous exploitons le score comme outil d'évaluation des solutions, il est important d'indiquer que le mode de classement induit obligatoirement le comportement de l'évalué. Le comportement du participant sera obligatoirement altéré s'il connaît le mode d'évaluation. Cette observation se vérifie dans le cas d'une personne physique ou d'un agent virtuel.

En effet, l'attribution de points au classement provient de différents critères. Ainsi, il est logique et compréhensible de privilégier, consciemment ou pas, les critères les plus pondérés.

Dans le cas de joueurs physiques, l'attention est immédiatement portée sur les critères prépondérants. Dans le cas d'agents virtuels, la sélection des bonnes solutions se fait selon les règles de scoring. Par conséquent seules les solutions satisfaisant au mieux les critères les plus pondérés seront conservées.

2.5 Modélisation des départements Production, Distribution et Finance

Nous nous positionnons au niveau de l'optimisation des décisions internes à l'entreprise. Comme nous l'avons présenté plus tôt, les trois départements Production, Distribution et Finance appellent une optimisation. La figure 2.11 nous permet de nous positionner visuellement par rapport à l'ensemble du problème que nous proposons de résoudre.

Ainsi, pour chaque hypothèse, nous souhaitons construire une solution optimale pour les départements Production, Distribution et Finance. Pour obtenir cet optimum, nous proposons de mettre en œuvre une résolution par un *programme linéaire mixte*. Nous ne décrirons, dans ce manuscrit, que les départements Production et Finance. Notre choix a été guidé par l'originalité des caractéristiques du département Production. Rappelons que ces caractéristiques sont imposées par les règles du jeu.

Le département Finance mérite aussi une section appropriée car il fait intervenir des notions économiques peu intégrées dans les modèles connus d'optimisation de production/supply chain management. Enfin, le département Distribution, assez proche du département Production sera donné en annexe.

Dans cette section, nous allons d'abord présenter le problème, les règles du jeu de chaque département et leurs spécificités. Nous proposerons ensuite une formulation en PLNE des contraintes et de la fonction objectif pour chaque département.

2.5.1 Notations

Les notations générales utilisées dans ce manuscrit sont listées dans le tableau 2.7. Notons que chaque machine m est dédiée au produit p . Cette machine sera du type t .

TABLEAU 2.7 – Notations générales

n	tour n (année n)
p	produit p
k	nombre de tours à optimiser
n_0	tour initial
m	machine m
z	marché z
$M_{p,t}$	ensemble des machines de type t pour le produit p

2.5.2 Le département Production

Rappel et précisions de certaines règles

Ce paragraphe permet de comprendre rapidement les règles du département Production.

En production, les participants peuvent s'orienter vers trois alternatives :

- **Production** : Investissement en machines, recrutement de personnel et maîtrise des moyens.
- **Sous-traitance** : Commande de produits finis à des sous-traitants spécialisés.
- **Mixte** : Investissement en machines et sous-traitance pour compléter la production.

La problématique du « make or buy » est donc particulièrement présente. Les sous-traitants externes au jeu proposent des produits finis à prix fixe.

Les machines, appelées lignes, peuvent être de deux types différents. Ces types ne peuvent pas coexister pour un même produit dans l'atelier. Si les participants décident de changer de type de lignes, toutes les machines de l'ancien type seront cédées. Les nouvelles machines arriveront alors avec des pénalités de production en première année correspondant au temps de mise en place.

Les deux types de machines diffèrent par leur prix, leur productivité, les ressources humaines nécessaires à leur bon fonctionnement, leur taux d'amortissement comptable (calcul de la dotation aux amortissements), la moins-value réalisée lors de leur cession et la capacité nominale de production. Enfin, les machines sont automatiquement vendues lorsqu'elles sont amorties en totalité. La productivité d'une machine dépend de son âge et de son type de façon non linéaire. Elle est définie de manière discrète pour chaque année d'utilisation. A l'opposé, nous considérons que l'amortissement comptable est linéaire ce qui rend identique la dotation aux amortissements pour tous les cycles d'exploitation de l'outil.

Pour ces multiples contraintes, nous avons défini un ensemble de variables pour chaque machine que nous intégrons dans l'optimisation. Cet ensemble de variables est défini quelque soit le type de machine acquise. Nous ajoutons à notre programme linéaire plus de machines que l'entreprise n'en dispose déjà. Ce surdimensionnement permet de pallier les décisions suivantes :

- Acquisition de nouveaux outils de production pour répondre à une augmentation de la demande,
- Changement de type d'outil de production,
- Remplacement des outils les plus vieux - les moins productifs - au profit de nouvelles lignes.

Quand une nouvelle machine est achetée, des frais de formation pour le personnel sont comptabilisés. Le nombre de travailleurs affectés à l'utilisation des machines est défini par type de machine. Pour encadrer les travailleurs, des cadres de production sont nécessaires. Le taux d'encadrement est défini par un paramètre spécifique général à l'entreprise.

Le directeur Production doit maîtriser son niveau de stock afin de soulager son besoin en fonds de roulement. Cet indicateur de maîtrise peut être défini par le département Finance ou Production. Deux autres choix sont possibles. Dans un premier temps, il est possible d'avoir recours aux heures supplémentaires. Les heures supplémentaires sont utilisées pour augmenter la capacité de l'atelier de production. Elles entraînent des frais supplémentaires indiqués dans le paramétrage. Elles sont également limitées en nombre. Cette capacité est fonction du nombre de salariés en poste. Dans un second temps, il est possible d'opter pour une commande en externe de produits finis qui s'ajustera automatiquement à un surplus de commandes. Cette option permet à l'équipe de se décharger des décisions d'approvisionnement. Elle permet ainsi de maximiser le taux de livraison. Cependant, cette souplesse entraîne des coûts complémentaires. Ces achats sont considérés comme des commandes exceptionnelles. Ils sont majorés par un taux défini dans le paramétrage initial.

Le département Production avait fait l'objet d'une étude partielle permettant d'offrir rapidement aux utilisateurs des outils simples pendant une session (PECH-GOURG et DUFOURNY [2012] et DUFOURNY et DHAENENS [2013]). De nombreuses contraintes n'avaient pas été intégrées. En particulier, l'exclusivité du mode de production était absent.

Modélisation du département Production

Les données du département Production sont listées dans le tableau 2.8. Ce tableau présente les grandeurs de production : temps de fabrication, productivité, ventes prévues... Pour mémoire, ces ventes prévisionnelles sont le fruit d'une analyse du marché et de la concurrence (Figure 2.11).

TABLEAU 2.8 – Paramètres de production

HV	Grande valeur arbitraire (high value)
K_p	Temps de production par produit p
$Nominal_capa_{p,t}$	Capacité de production nominale d'une machine de type t pour p
$Productivity_{p,t,i}$	Productivité d'une machine de type t pour p de i ans
Cst_{OT}	Heures supplémentaires (Over Time) autorisées par les règles du jeu
$Init_stock_{p,n}$	Stock initial de p au tour n
$Sales_{p,n}$	Ventes prévisionnelles de p au tour n (en nombre de produits)

Afin de répondre aux ventes prévisionnelles nécessaires, il faut définir le nombre de machines à acheter, céder, le nombre de produits finis à acheter aux sous-traitants. Le tableau 2.9 indique les variables de décisions de production, pour chaque produit p au tour n .

TABLEAU 2.9 – Variables de production pour un produit p au tour n

$NbMachine_{p,n}$	Nombre de machines disponibles
$Invest_{p,n}$	Nombre de machines à acheter
$Transfer_{p,n}$	Nombre de machines à vendre
$isType_{p,n,t}$	= 1 si le type de machine est de type t (0 sinon)
$isChanged_{p,n}$	= 1 si l'on change de type de machine (0 sinon)
$Production_{p,n}$	Quantité de produits à fabriquer
$SC_{p,n}$	Nombre de produits achetés
$SCEXtra_{p,n}$	Nombre de produits achetés en express
$EndStock_{p,n}$	Nombre de produits en stock en fin de tour
$OT_{p,n}$	= 1 si les heures supplémentaires sont possibles (0 sinon)
$ProductionOT_{p,n}$	Nombre de pièces produites pendant les heures supplémentaires

Afin de calculer la capacité de production, chaque machine m est décrite par l'ensemble de variables reprises dans le tableau 2.10 au tour n :

Le paragraphe suivant présente les contraintes de production à remplir pour chaque hypothèse considérée (début de construction d'une solution). La fonction objectif sera présentée en fin de paragraphe.

Programme linéaire en nombres entiers pour le département Production

Ce modèle a fait l'objet d'un article. Il a été présenté à la conférence MIM 2016 (DUFOURNY et DHAENENS [2016]). Le principal but de la production est d'être capable de produire suffisamment pour assurer la livraison des clients et ainsi maximiser le taux de service.

TABLEAU 2.10 – Variable des machines

$IsUsed_{m,n}$	= 1 si la machine m est utilisée (0 sinon)
$isAged_{m,n,i}$	= 1 si l'âge de la machine m est i (0 sinon)
$isBought_{m,n}$	= 1 si la machine m est achetée (0 sinon)
$isTransfer_{m,n}$	= 1 si la machine est vendue (0 sinon)
$ProductionCapa_{m,n}$	Capacité de production de la machine m au tour n
$MNV_{m,age,t}$	Valeur Nette Comptable de la machine m , de type t âgée de age
$CapitalLoss_{m,n}$	Moins value de cession pour la machine m

Cette contrainte peut être exprimée comme suit :

$$\forall p, \forall n \geq 2, \quad \begin{aligned} & EndStock_{p,n-1} \\ & + Production_{p,n} \\ & + ProductionOT_{p,n} \\ & + SC_{p,n} \\ & + SCExtra_{p,n} \end{aligned} = Sales_{p,n} + EndStock_{p,n} \quad (2.1)$$

Il n'y a pas de limitation en achat express ou en approvisionnement par des sous-traitants externes. La planification de production est limitée par les performances des machines et par leur nombre. Le planning de production peut être considéré comme étant la somme des capacités de production unitaires des machines :

$$\forall n, \forall p, \forall t, \quad ProdCapacity_{p,n,t} = \sum_{m \in M_{p,T}} ProductionCapa_{m,n,t} \quad (2.2)$$

Le type de machine est exclusif par référence et la capacité de production en dépend :

$$\forall n, \forall p, \quad \sum_t isType_{p,n,t} = 1 \quad (2.3)$$

L'équation (2.4) permet de n'intégrer à la capacité de production que les lignes du type t sélectionné.

$$\forall n, \forall p, \forall t, \quad ProdCapacity_{p,n} \leq ProductionCapa_{p,n,t} + HV * \sum_{t' \neq t} isType_{p,n,t'} \quad (2.4)$$

La production d'un produit p au cycle n ne pourra pas dépasser la capacité de production totale comme indiqué dans l'équation (2.5).

$$\forall n, \forall p, \quad Production_{p,n} \leq ProdCapacity_{p,n} \quad (2.5)$$

La production réalisée par les heures supplémentaires est limitée à une fraction de la production réelle. Cst_{OT} représente une constante définie dans l'univers économique qui correspond au taux maximal d'heures supplémentaires réalisables par salarié.

$$\forall n, \forall p, \quad ProductionOT_{p,n} \leq ProdCapacity_{p,n} * Cst_{OT} \quad (2.6)$$

Un booléen, $isChanged_{p,n}$ est utilisé afin de détecter si le type de ligne de production a été modifié (Contraintes (2.7) et (2.8)).

$$\forall p, \forall t, \forall n \geq 2, \quad isChanged_{p,n} \geq IsType_{p,n-1,t} - IsType_{p,n,t} \quad (2.7)$$

$$\forall p, \forall t, \forall n \geq 2, \quad isChanged_{p,n} \geq IsType_{p,n,t} - IsType_{p,n-1,t} \quad (2.8)$$

Le nombre de machines disponibles impacte la capacité de production. Nous la déterminons donc de façon indépendante par machine comme suit :

$$\forall m, \forall n, \quad IsUsed_{m,n} = \sum_{j=1}^{n-1} isBought_{m,j} - \sum_{j=1}^n isTransfer_{m,j} \quad (2.9)$$

Chaque machine ne peut être vendue qu'une seule fois ($n \in [n_0, \dots, n_0 + k]$) :

$$\forall m, \quad \sum_{j=1}^{n_0+k} isTransfer_{m,j} \leq 1 \quad (2.10)$$

Chaque machine ne peut être achetée qu'une seule fois :

$$\forall m, \quad \sum_{j=1}^{n_0+k} isBought_{m,j} \leq 1 \quad (2.11)$$

Toutes les valeurs de la variable $isAged$ prennent 0 si la machine est vendue.

$$\forall m, \forall n, \quad \sum_{i=1}^{n_0+k} isAged_{m,n,i} \leq 1 - isTransfer_{m,n} \quad (2.12)$$

$$\forall m, \forall n, \forall i, \forall (n - i + n_0 > 0), \quad isAged_{m,n,i} = isBought_{m,n-i+n_0} - \sum_{k=1}^{n+n_0} isTransfer_{m,k} \quad (2.13)$$

$$\forall m, \forall i, \forall j, \quad \forall (i - j + n_0 \leq 0), \quad isAged_{m,i,j} = 0 \quad (2.14)$$

La capacité de production dépend :

- d'un critère de cycle de vie : plus de production après amortissement complet (2.15)
- d'une utilisation partielle de l'outil de production si le type de celui-ci est modifié en cours de période : la règle du jeu impose de changer l'ensemble du parc si l'on change le type des lignes de production (2.16)
- d'un critère d'activité (2.17)
- d'un critère d'obsolescence (2.18)

$$ProductionCapa_{m,n} \leq NominalCapa_{p_m,t_m} * \sum_i \left(Productivity_{p_m,t_m,i} * isAged_{m,n,i} \right) \quad (2.15)$$

$$ProductionCapa_{m,n} \leq \frac{NominalCapa * Productivity_{p_m,t_m,1} * \frac{10}{12}}{+HV * (1 - isChanged_{p_m,n})} \quad (2.16)$$

$$ProductionCapa_{m,n} \leq IsUsed_{m,n} * HV \quad (2.17)$$

$$ProductionCapa_{m,n} \leq VNC_{m,n} * HV \quad (2.18)$$

Toutes ces contraintes nous permettent d'assurer une production répondant aux besoins des clients tout en respectant les règles du jeu.

TABEAU 2.11 – Coûts intégrés dans le programme linéaire de Production

Ct_{RM}	Coût des matières premières (Raw Materials)
Ct_{HR}	Salaires du personnel en production
Ct_{OT}	Coût des heures supplémentaires
Ct_{Amort}	Dotation aux amortissements
$Ct_{Training}$	Coût de formation
$Capital_{Loss}$	Moins-value pour cession d'actifs
Ct_{Stocks}	Coûts de stocks
Ct_{ForeM}	Coûts de l'encadrement de production
Ct_{Adm}	Coûts administratifs

La fonction objectif

Le but de la fonction objectif est de minimiser l'ensemble des coûts de production. Les coûts de production sont listés dans le tableau 2.11.

La fonction objectif peut alors être exprimée par :

$$\text{Min} \left(\begin{array}{c} Ct_{RM}_{p,n} + \\ Ct_{Hr}_{p,n} + \\ CT_{OT}_{p,n} + \\ \sum_{n=1}^k \sum_p Ct_{Amort}_{p,n} + \sum_n CT_{ForeM}_n + \\ Ct_{Training}_{p,n} + \\ Capital_{Loss}_{p,n} + \\ Ct_{Stocks}_{p,n} \\ Ct_{Adm}_n \end{array} \right) \quad (2.19)$$

Chaque coût est obtenu par les équations suivantes :

Le coût des **Matières premières** est obtenu avec l'équation (2.20).

$$\forall n, \forall p, \quad Ct_{RM}_{p,n} = \begin{array}{l} A_p * (Production_{p,n} + ProductionOT_{p,n}) \\ + B_p * SC_{p,n} \\ + C_p * SCE_{p,n} \end{array} \quad (2.20)$$

avec A_p , B_p et C_p des données du marché qui permettent pour le produit p de définir respectivement le coût des matières premières, le coût de la sous-traitance et enfin le coût de la sous-traitance des produits commandés en express.

La production réalisée avec des heures supplémentaires ($ProductionOT$) est obtenue par :

$$\forall n, \forall p, \quad ProductionOT_{p,n} = \frac{OT_{p,n}}{K_p} \quad (2.21)$$

La **masse salariale de production** est obtenue par l'équation (2.22).

$$\forall n, \quad Ct_{HR}_n = \sum_p Ct_{HR}_{p,n} \quad (2.22)$$

$$\forall n, \forall p, \forall t, \quad Ct_{HR}_{p,n} \geq NbMachine_{p,n} * E_t - HV * \sum_{t' \neq t} IsType_{p,n,t'} \quad (2.23)$$

et E_t est une constante du marché.

avec

$$\forall n, \forall p, \forall t, \quad NbMachine_{p,n} \geq \sum_{m \in M_{p,t}, i < n} isBought_{m,i} - \sum_{M_{p,t}, i \leq n} isTransfer_{m,i} - HV * \sum_{t' \neq t} IsType_{p,n,t'} \quad (2.24)$$

Le coût des **heures supplémentaires** provient de l'équation (2.25).

$$\forall n, \quad Ct_{OT}_n = \sum_p Ct_{OT}_{p,n} \quad (2.25)$$

avec

$$\forall n, \forall p, \forall t, \quad Ct_{OT}_{p,n} \geq Nb_{OT}_{p,n} * F_t - HV * \sum_{t' \neq t} IsType_{p,n,t'} \quad (2.26)$$

Le coût de **l'encadrement de production** est obtenu par l'équation (2.27).

$$\forall n, \quad Ct_{ForeM}_n \geq NbForeM_n * Cst \quad (2.27)$$

avec

$$\forall n, \forall p, \forall t, \quad NbWorker_{p,n} \geq NbMachine_{p,n} * Cst_t - HV * \sum_{t' \neq t} IsType_{p,n,t'} \quad (2.28)$$

$$\forall n, \quad NbWorker_n = \sum_p NbWorker_{p,n} \quad (2.29)$$

$$\forall n, \quad NbForeM_n \geq \frac{NbWorker_n}{MgtCst} \quad (2.30)$$

et $MgtCst$ une constante du scénario.

Le **coût des stocks** est obtenu par l'équation (2.31).

$$\forall n, \forall p, \quad Ct_{Stocks}_{p,n} = V * \sum_p End_{Stock}_{p,n} \quad (2.31)$$

et V est le coût d'un produit en stock.

En cas de cession d'une machine, la **moins value de cession** provient de l'équation (2.32).

$$\forall n, \quad Capital_{loss}_n = \sum_p Capital_{loss}_{p,n} \quad (2.32)$$

avec

$$\forall n, \forall p, \quad Capital_{loss}_{p,n} = \sum_{M_{p,t}} Capital_{loss}_{m,n} \quad (2.33)$$

$$\forall m, \forall n, \quad MNV_{m,n} = \sum_i MNV_{m,age,t} * isAged_{m,n,i} \quad (2.34)$$

$$\forall m, \forall n, \quad Capital_{loss}_{m,n} \geq MNV_{m,n} * K_m - HV * (1 - isTransfer_{m,n}) \quad (2.35)$$

avec K_m une constante issue du paramétrage du jeu. Cette constante représente le taux de reprise d'une machine m par rapport à sa valeur nette comptable.

$$\forall n, \forall p, \forall t, \quad Capital_loss_{p,n} \geq Capital_loss_{p,n,t} - HV * \sum_{t' \neq t} IsType_{p,n,t'} \quad (2.36)$$

Le vieillissement comptable des machines est matérialisé par la **dotation aux amortissements** qui est obtenue par l'équation (2.37).

$$\forall n, \quad Ct_Amort_n = \sum_p Ct_Amort_{p,n} \quad (2.37)$$

$$\forall n, \forall p, \forall t, \quad Ct_Amort_{p,n} \geq NbMachine_{p,n} * Cst_t - HV * \sum_{t' \neq t} IsType_{p,n,t'} \quad (2.38)$$

Les **coûts administratifs** s'obtiennent rapidement par l'équation (2.39).

$$\forall n, \quad Ct_Adm_n \geq \sum_p Ct_RM_{p,n} * Cst \quad (2.39)$$

Enfin, lors de l'acquisition de nouvelles machines, les **frais de formation** s'obtiennent par l'équation (2.40).

$$\forall n, \quad Ct_Training_n = \sum_p Ct_Training_{p,n} \quad (2.40)$$

avec

$$\forall n, \forall p, \forall t, \quad Ct_Training_{p,n} \geq \sum_{M_p} isBought_{m,n} * Cst_t - HV * isType_{p,n,t'} \quad (2.41)$$

Optimisation du modèle

Pour améliorer le processus d'optimisation de solutions et éviter la génération de solutions équivalentes avec différents indices de machines, nous ajoutons la contrainte (2.42). Nous forçons ainsi l'optimisation à utiliser les machines disponibles de plus faible indice (2.43).

$$\forall t, \forall n, \forall p, \quad \sum_{m \in M_{p,t}} \left(\begin{array}{c} \sum_{i \leq n} isBought_{m,i} \\ - \sum_{i \leq n} isTransfer_{m,i} \end{array} \right) \leq isType_{p,n,t} * HV \quad (2.42)$$

$$\forall m, \forall n, \quad isBought_{m,n} \leq \sum_{i \leq n} isBought_{m-1,i} \quad (2.43)$$

La contrainte (2.44) supprime toutes les solutions dupliquées. Elle permet donc de réduire considérablement le temps de calcul. De plus, elle supprime de la résolution les machines dont le type ne correspond pas au type actuel.

$$\forall n, \forall p, \quad \sum_{m \in M_{p,t}} \left(\begin{array}{c} \sum_{i \leq n-1} isBought_{m,i} \\ - \sum_{i \leq n} isTransfer_{m,i} \end{array} \right) \leq (1 - isChanged_{p,n}) * HV \quad (2.44)$$

2.5.3 Modélisation du département Distribution

Les points de vente ou points de distribution remplissent deux rôles :

- **Inciter** : plus les entreprises disposent d'une forte capacité de distribution et plus leurs chances de vendre augmentent. Cet aspect est concurrentiel et correspond à la densité des distributeurs sur le marché.
- **Réguler** : la livraison des clients impose une capacité de distribution suffisante exprimée en chiffre d'affaires. Si cette capacité est insuffisante, les entreprises resteront avec du stock de produits finis. Ce stock pourra entraîner des problèmes de solvabilité à court terme. Cette non-livraison impactera également le taux de service. Cet indicateur est intégré au classement/score de l'entreprise. Il revêt donc une importance particulière.

Il existe deux types de point de vente. Chaque type (h) présente des caractéristiques distinctes. Le premier est un distributeur qui est la propriété de l'entreprise. Son coût est donc essentiellement constitué de charges fixes correspondant à des frais de personnel. Le second est pour sa part assimilable à un agent commercial rémunéré en fonction de sa performance (donc variable) mesurable au chiffre d'affaires généré. Cependant, les frais de rupture de contrat sont très élevés pour ce mode de distribution.

Chaque type de point de vente offre une capacité de distribution spécifique et invariable dans le temps.

La multiplicité des types de points de vente impose aux participants le calcul du seuil d'indifférence. Ce seuil est exprimé en chiffre d'affaires ou en nombre de pièces. Il représente la limite à partir de laquelle il est plus rentable de distribuer ses produits en implantant ses propres succursales. En deçà de cette limite, le recours aux agents commerciaux est recommandé.

Pour les mêmes raisons que dans la partie consacrée à la production, nous avons décidé de définir chaque point de distribution individuellement. Tous les points seront caractérisés par une zone (z), un type et offriront une capacité de distribution *SalesCapa*.

Pour dimensionner le nombre de points de distribution du modèle, nous considérons :

- Le nombre de points de vente actuels avant l'optimisation,
- Les prévisions de vente,
- La capacité de distribution pour chaque type,
- Les coûts liés à chaque type.

Le programme linéaire détaillé se trouve en annexe car il ne présente pas de difficulté majeure. Il est en effet très proche du programme linéaire proposé pour le département Production.

Fonction objectif de la distribution

Nous avons souhaité réduire au maximum les coûts de distribution considérés. Les coûts que nous avons identifiés sont :

2.5.4 Modélisation du département Finance

Rappel et précisions de certaines règles

A chaque tour, les participants ont pour mission de financer leur croissance. En général, le scénario économique préétabli prévoit des croissances de marché importantes.

TABLEAU 2.12 – Coûts intégrés dans le programme linéaire de Distribution

<i>Ct_Dismiss</i>	Coût de rupture de contrat
<i>Ct_Amort</i>	Dotation aux amortissements
<i>Ct_Fixed</i>	Coût fixe
<i>Ct_Variable</i>	Coût variable indexé sur les ventes (chiffre d'affaires)
<i>Capital_Loss</i>	Moins value de cession

Les équipes démarrent toutes sans outil de production et ont recours uniquement à la sous-traitance. Enfin, afin de se différencier, les participants mettent en œuvre des plans de recherche et développement. Ainsi, les équipes doivent financer leur croissance.

On distingue différents types de ressources financières. Les entreprises peuvent solliciter des ressources financières à long terme. Dans ce cas, elles optent pour l'appel aux actionnaires et/ou le recours à l'emprunt. Ces opérations sont limitées pour chaque tour de jeu. La limite de capitalisation est donnée par le système au début d'un tour. La souscription à de nouveaux emprunts entraînera une modification du ratio d'endettement. Celui-ci est sous contrôle. Un taux d'endettement supérieur à la valeur spécifiée dans les paramètres entraînera une pénalité sur le score de l'équipe. Les emprunts peuvent être contractés sur moyen ou long terme et à des taux d'intérêts variables ou fixes. Le taux des emprunts dépend d'un taux interbancaire classique intégré au scénario de marché pour une part. Les performances des entreprises impactent aussi cette grandeur.

Sur le court terme, les entreprises ont la possibilité de modifier leur délai de paiement fournisseur et de distribuer des dividendes. Ce dernier point accroîtra l'attractivité de l'entreprise et autorisera une enveloppe d'augmentation de capital plus élevée sur les tours suivants.

Une attention toute particulière sera donnée à la notion de solvabilité court terme et long terme. En effet, en cas d'insolvabilité, les entreprises seront pénalisées et enregistreront des incidents. En cas d'incidents répétés, elles seront déclassées et considérées en faillite ou défaillance. Les limites de découvert autorisé sont calculées automatiquement par le modèle en analysant le besoin en fonds de roulement. Cette limite augmente en fonction du compte client des entreprises et des stocks mais diminue en contrepartie du compte fournisseur et des dettes d'état (Impôt sur les sociétés).

Pour la modélisation, nous avons choisi de simplifier le mode de calcul. L'outil Finance doit donner avant tout une réponse booléenne : les besoins financiers issus des départements Production et Distribution sont-ils finançables ? Si oui, quelles décisions devraient être prises pour réduire le coût du financement ?

La faisabilité financière est, en partie, évaluée par le taux d'endettement maximal. Ce taux d'endettement représente le taux de dépendance financière de l'entreprise vis-à-vis de ses établissements financiers. Il est le ratio de l'endettement bancaire (long terme et court terme) divisé par les capitaux propres de l'entreprise. Cette faisabilité est aussi validée après analyse du niveau de trésorerie. Celui-ci doit rester supérieur au découvert autorisé par l'établissement bancaire. Ce découvert autorisé est lui-même fonction du besoin en fonds de roulement de l'entreprise. Les stocks et les créances clients accroissent le niveau de découvert acceptable. Les dettes d'exploitation court terme ont tendance à réduire cette autorisation.

Ainsi, seules des données du cycle en cours sont évaluées. Par conséquent, l'optimisation choisie n'intègre que le tour actuel de jeu.

Notations

Les données du département Finance sont :

α	Durée de l'emprunt (modalité)
β	Taux de l'emprunt (modalité)
$EndCash_0$	Trésorerie initiale (après intégration des décisions de la production et distribution)
$Loan_0$	Emprunts restants dûs avant optimisation
$Equity_0$	Capital initial (avant optimisation)
Max_{debt}	Ratio d'endettement maximal
$Capa_{Recap}$	Recapitalisation maximale
$Capa_{Debt}$	Emprunt maximal autorisé
BO_{max}	Découvert maximal autorisé
$Rate_{BO}$	Coût du découvert
$Am_{\alpha,\beta}$	Ratio correspondant au capital amorti de l'emprunt la première année
GPC_{Base}	Délai de paiement fournisseur
$wACC$	Coût moyen du capital

Les objectifs du département sont de définir les meilleures décisions financières pour permettre la croissance de l'entreprise au coût le plus faible possible.

Les variables financières sont :

$Loan_{\alpha,\beta}$	Emprunt pour la durée α et au taux β
$Recap$	Augmentation de capital
GPC	Délai de paiement fournisseur
$Dividend$	Distribution de dividendes
$EndCash$	Trésorerie finale
$NegEndCash$	Valeur absolue de la trésorerie finale si celle-ci est négative
$isCashPlus$	= 1 si la trésorerie est positive (0 sinon)

Modèle linéaire du département Finance

Le principal objectif est d'éviter des incidents bancaires. Ainsi, il est important de s'assurer que la trésorerie finale sera bien supérieure au découvert maxi autorisé par la banque (2.45) et de surveiller le ratio d'endettement long terme (2.46).

$$EndCash \geq BO_{max} \quad (2.45)$$

$$\begin{pmatrix} \sum_{\alpha,\beta} Loan_{\alpha,\beta} \\ + Loan_0 \\ + NegEndCash \end{pmatrix} \leq \begin{pmatrix} Equity_0 \\ - Dividend \\ + Recap \end{pmatrix} * Max_{debt} \quad (2.46)$$

Le solde de trésorerie final est anticipé par l'équation suivante :

$$EndCash \leq \begin{aligned} & EndCash_0 \\ & + \sum_{\alpha,\beta} Loan_{\alpha,\beta} * (1 - Am_{\alpha,\beta}) \\ & + Recap - Dividend \\ & - \frac{GPC - GPC_0}{365} * Cst_{Achat} \end{aligned} \quad (2.47)$$

avec GPC_0 , le délai de paiement moyen attendu par les fournisseurs.

Ce solde intègre le solde initial, l'appel à de nouveaux emprunts, les éventuelles augmentations de capital et de distribution de dividendes. Il présente également la décision de délai de paiement des fournisseurs. La constante Cst_{Achat} reprend l'ensemble des achats externes :

- les coûts d'approvisionnements
- les coûts de publicité
- les coûts de transport
- les coûts des options

L'entreprise doit respecter un appel de fonds maximal :

$$\sum_{\alpha,\beta} Loan \leq Capa_{Debt} \quad (2.48)$$

$$Recap \leq Capa_{Recap} \quad (2.49)$$

Fonction objectif du département Finance

Nous cherchons à minimiser les coûts suivants :

Ct_{Loan}	Coût des emprunts
Ct_{Recap}	Coût du retour sur investissement des actionnaires
$Ct_{Overdraft}$	Coût du découvert bancaire
Ct_{GPC}	Coût de l'escompte des fournisseurs

La fonction objectif est donc la suivante :

$$Min \left(\begin{array}{c} Ct_{Loan}+ \\ Ct_{Recap}+ \\ Ct_{Overdraft}+ \\ Ct_{GPC} \end{array} \right)$$

Le **coût des emprunts** est calculable par la fonction suivante. Il s'agit d'une valeur approchée. Nous avons souhaité favoriser ce mode de calcul car il permet de maintenir le modèle linéaire.

$$Ct_{Loan} \geq \sum_{\alpha,\beta} (Loan_{\alpha,\beta} * Rate_{\alpha,\beta}) \quad (2.50)$$

Le **coût de l'augmentation de capital** s'obtient par la formule suivante. Notons ici que nous avons fait une estimation du coût du capital $wACC$ simplifiée une nouvelle fois. Ce coût est assimilé à une rémunération attendue de l'actionnaire pour son risque.

$$Ct_{Recap} \geq Recap * wACC \quad (2.51)$$

Le **coût du découvert bancaire** se calcule par :

$$Ct_{Overdraft} \geq NegEndCash * Rate_{BO} \quad (2.52)$$

Afin de déterminer ce coût, nous devons préalablement évaluer le découvert bancaire (la trésorerie négative) (2.53), (2.54). Ensuite, le taux du découvert est appliqué afin d'en déduire son coût (2.55), (2.56).

$$EndCash - HV * isCashPlus \leq 0 \quad (2.53)$$

$$EndCash + HV * (1 - isCashPlus) \geq 0 \quad (2.54)$$

$$NegEndCash \geq -EndCash - HV * isCashPlus \quad (2.55)$$

$$NegEndCash \geq 0 \quad (2.56)$$

Coût de l'escompte des fournisseurs

$$Ct_GPC \geq Cst * (GPC_{base} - GPC) \quad (2.57)$$

Notons que les fournisseurs souhaitent un paiement défini à GPC_{base} jours fin de mois. Une décision à ce niveau n'entraîne donc aucun coût. GPC_{base} est une constante du scénario.

2.5.5 Mise en œuvre

Choix de la plateforme technique

Pour la résolution des programmes linéaires présentés, nous avons choisi la bibliothèque *Lp_solve* (EIKLAND [2013]). *Lp_solve* est une bibliothèque libre de droits de résolution de programmes linéaires mixtes basée sur la méthode du simplexe révisée et la méthode Branch-and-Bound pour les entiers. *Lp_solve* offre un ensemble de routines externes qui peuvent être appelées par des programmes indépendants sur toutes plateformes.

La plateforme CPLEX, plus connue, est très largement utilisée. Elle est certainement plus performante que la librairie choisie. Sur nos essais, les temps d'exécution étaient raisonnables et surtout compatibles avec la finalité de cette thèse. Nous avons limité le processus d'optimisation à 90 secondes afin de rester dans un temps de calcul acceptable au risque de perdre quelques solutions.

Aussi, l'exploitation à des fins commerciales de CPLEX est onéreuse. Cet aspect financier la rend incompatible avec l'exploitation prévue du processus d'optimisation pour une première mise en place commerciale. Nous avons donc préféré réaliser nos essais sur la librairie *Lp_solve*.

Parallélisation de l'exécution

Pour atteindre une résolution rapide, plusieurs hypothèses sont optimisées en parallèle. A l'intérieur de chaque processus de construction de solution, les départements Production et Distribution sont également parallélisés. Nous avons représenté cette parallélisation sur la figure 2.12.

Le processus d'optimisation débute par la génération des hypothèses par recherche locale. Ensuite, les p solutions issues de ces hypothèses vont être construites indépendamment. Ces constructions sont simultanées.

Dans notre figure, nous avons représenté, comme dans UpStrat, pour chaque hypothèse, trois produits et trois marchés. Les optimisations des départements sont aussi parallélisées. Notons que le temps d'exécution de chaque processus est différent. C'est pourquoi, l'optimisation du département Finance démarre après le dernier processus terminé.

Le temps de construction de chaque solution, représenté à gauche de la figure, est lui aussi différent. Le tournoi de sélection de la meilleure solution ne peut s'opérer qu'après la finalisation de la construction de la solution la plus longue.

Nous avons intégré des points de synchronisation afin de valider le processus complet.

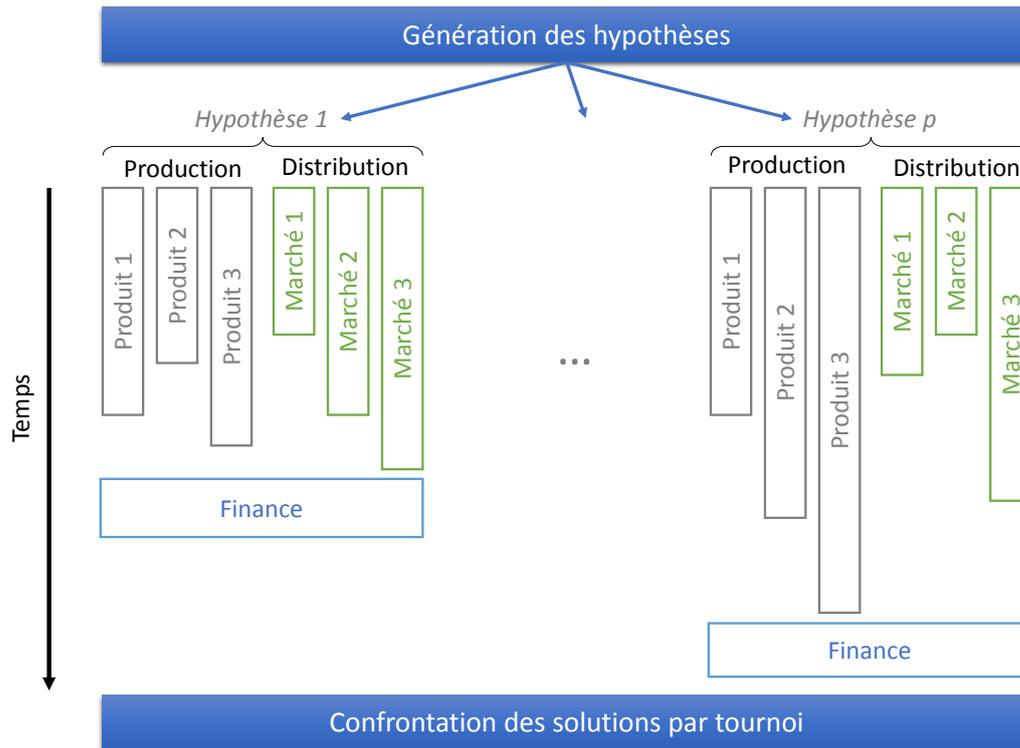


FIGURE 2.12 – Parallélisation des optimisations

Performance des modèles linéaires

Les modèles linéaires présentés dans ce chapitre offrent des résultats excellents en comparaison de décisions d'équipes. Ils sont bien souvent meilleurs en terme de réduction de coûts. Ils permettent également de prendre des décisions optimisées sur plusieurs cycles.

Aussi, ils peuvent être sollicités indépendamment par le joueur. Cette aspect nous confère de la souplesse lors de l'utilisation en mode *coach*.

Enfin, la suppression des solutions non réalisables en fonction des contraintes intégrées nous assure une efficacité de la résolution.

2.6 Conclusions

Dans ce chapitre, nous avons dans un premier temps posé les bases théoriques en stratégie et en comptabilité de gestion nous permettant de valider nos choix ultérieurs de modélisation.

Dans un deuxième temps, nous avons observé le comportement de joueurs lors de leurs prises de décisions. Ces observations nous ont permis de schématiser ce processus parfois complexe. Nous avons donc ensuite proposé d'adapter ce processus à un agent autonome. Nous avons également mis en œuvre une méthode de sélection de la meilleure solution dans le contexte UpStrat.

Dans un troisième temps, nous avons modélisé les départements Production, Distribution et Finance. Ces optimisations nous ont permis de construire des solutions réalisables et cohérentes à partir des ventes prévisionnelles définies.

Nous avons exprimé notre choix de plateforme technique et validé nos déploiements

au regard de décisions de joueurs réels.

Afin de construire les solutions, nous avons proposé une hybridation des méthodes d'optimisation. Nous avons déployé des méthodes exactes (programme linéaire mixte) et avons proposé une méthode approchée sous la forme d'une recherche locale pour la génération des hypothèses.

Nous proposons dans le chapitre suivant de nous focaliser sur cette approche par recherche locale, point de départ d'une solution.

Ainsi, nous introduisons l'optimisation combinatoire et présentons en détail la méthode utilisée. En fin de chapitre, nous paramétrons cette méthode afin de la rendre rapide et efficace.

Chapitre 3

Génération des hypothèses par méthode d'optimisation

L'objectif de ce chapitre est de présenter la méthode que nous proposons pour permettre à l'agent de construire un bon ensemble de décisions ou plus précisément d'hypothèses qui serviront à la construction d'une solution. Ce travail a fait l'objet d'une présentation à la conférence MIC'15 (Metaheuristics International Conference) (DUFOURNY et DHAENENS [2015]).

Comme présenté dans le chapitre 2, nous traduirons ce problème en un problème d'optimisation combinatoire que nous avons décomposé en différents sous-problèmes (voir figure 2.11). Nous nous concentrerons dans ce chapitre sur la partie « génération de l'hypothèse ».

Afin de valider l'approche proposée, nous considérerons dans un premier temps que les décisions des équipes concurrentes sont connues de l'agent. Cette situation est volontairement utopique et injuste. Cependant, ce cadre nous permettra une estimation des ventes prévisionnelles correcte sur lesquelles sont basées les optimisations suivantes. Dans le chapitre 4, nous proposerons une méthode d'anticipation des décisions des concurrents à l'aide de méthodes d'extraction de connaissances. Ces méthodes nous repositionneront dans un contexte classique sans connaissance des décisions réelles des joueurs concurrents. Nous y respecterons donc la confidentialité des décisions et un jeu concurrentiel juste et équitable.

Afin de permettre aux non spécialistes du domaine d'avoir les éléments nécessaires à la compréhension de l'approche proposée, ce chapitre va dans un premier temps apporter quelques éléments sur les problèmes d'optimisation combinatoire et leurs méthodes de résolution.

Dans un second temps, nous présenterons la méthode d'optimisation que nous proposons pour la génération d'une bonne hypothèse (bon ensemble de décisions). Ensuite, nous relaterons les expérimentations menées afin de déterminer un paramétrage performant.

Enfin, la méthode paramétrée sera testée lors d'une mise en concurrence au sein de parties réelles afin d'en évaluer les performances.

3.1 Introduction à l'optimisation combinatoire

3.1.1 Définition d'un problème d'optimisation combinatoire

Les problèmes d'optimisation combinatoire sont très présents dans notre quotidien. Citons l'exemple du problème du voyageur de commerce (Traveling Salesman Problem (TSP)) qui doit passer par n villes une seule fois et revenir à son point de départ tout en minimisant la distance totale de son trajet.

Ce problème est de complexité élevée $(n - 1)!$. Ainsi, il devient impossible de résoudre des problèmes de quelques dizaines de villes si l'on recherche la meilleure solution parmi toutes les solutions possibles (toutes les permutations de n éléments).

Cependant différentes méthodes d'énumération intelligente permettent d'éviter l'énumération exhaustive. Ces méthodes couplées aux progrès technologiques font que des instances de ce problème ayant plusieurs milliers de villes peuvent être résolues.

De façon générale, la résolution d'un problème d'optimisation combinatoire consiste à trouver la ou les meilleures solutions parmi un ensemble fini de solutions, pouvant être de très grande taille, répondant à la problématique de maximisation ou de minimisation d'un critère sous un ensemble de contraintes.

Dans le cas d'une minimisation, nous modélisons de manière générale ce type de problème par l'équation suivante (3.1)

$$\min_{s \in \mathcal{D}} f(s) \tag{3.1}$$

où

- \mathcal{D} représente l'ensemble des solutions appelé *espace de recherche*.
- f une fonction objectif qui, pour chaque solution s , affecte une valeur (la plupart du temps réelle).

Cette modélisation est applicable à bon nombre de situations en planification, télécommunication...

3.1.2 Modéliser un problème d'optimisation combinatoire

Afin de répondre aux importants besoins d'optimisation de moyens, de nombreuses méthodes d'optimisation en recherche opérationnelle ont été développées. Cependant, avant de sélectionner la méthode la plus appropriée au problème rencontré, il faut :

- définir l'ensemble des solutions réalisables : Qu'est ce qu'une solution réalisable ?
- définir la fonction objectif à optimiser : Quel est l'objectif de l'optimisation ?

Afin d'identifier le plus fidèlement possible l'ensemble des solutions réalisables, il faut avant toutes choses, lister les contraintes auxquelles les solutions doivent répondre. Cette étape impose la vision d'un expert qui connaît particulièrement le problème.

La définition de la fonction objectif est réalisée également en accord avec l'expert. Il sera capable de définir le résultat attendu et donc d'exprimer explicitement ou implicitement la ou les ressources à optimiser. Ces ressources peuvent être de nombreux types : ressources monétaires, ressources temporelles, ressources humaines...

Il n'est d'ailleurs pas toujours possible de trouver un objectif unique, le problème peut alors être modélisé comme un problème multi-objectif.

3.1.3 Principales méthodes d'optimisation

Les méthodes d'optimisation sont divisées en deux grandes familles. La première famille regroupe les méthodes exactes. Ces méthodes sont principalement basées sur une énumération intelligente. Elles donnent une garantie de trouver la solution optimale pour une instance de taille finie dans un temps limité et de prouver son optimalité (PUCHINGER et RAIDL [2005]). Cependant, leur usage est réservé aux problèmes de petite taille.

La seconde famille regroupe, pour sa part, les méthodes approchées (ou heuristiques). En optimisation combinatoire, une heuristique est un algorithme approché qui permet d'identifier en temps polynomial au moins une solution réalisable rapide, pas obligatoirement optimale. L'usage d'une heuristique est efficace pour calculer une solution approchée d'un problème et ainsi accélérer le processus de résolution exacte. Généralement une heuristique est conçue pour un problème particulier, en s'appuyant sur sa structure propre sans offrir aucune garantie quant à la qualité de la solution calculée. Ces heuristiques sont très adaptées aux problèmes de grande taille.

La figure 3.1 permet d'avoir un aperçu et une classification des différentes méthodes d'optimisation que l'on retrouve généralement dans la littérature.

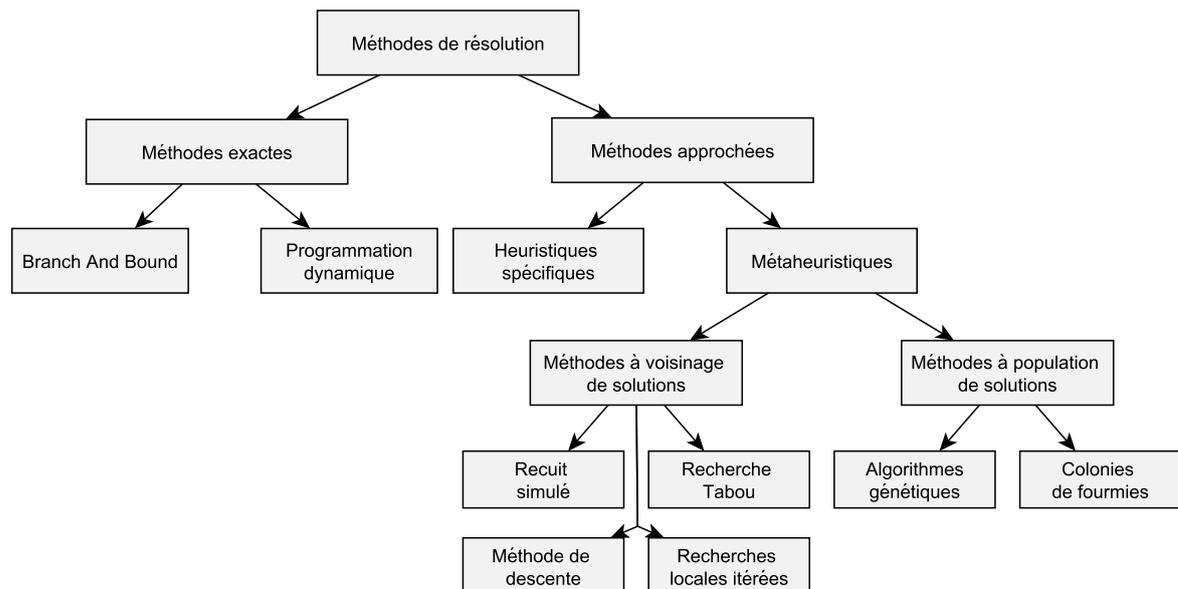


FIGURE 3.1 – Classement des principales méthodes d'optimisation

Si nous nous trouvons dans un cas où le nombre de solutions envisageables est raisonnable, une solution générée par une méthode exacte sera utilisée. Cette méthode offrira la meilleure solution au problème. Cette solution sera *l'optimum global*.

Cependant, ces méthodes exactes ne sont plus exploitables dès que la taille du problème augmente. Les temps de calcul deviennent trop longs. Les méthodes approchées constituent alors une alternative intéressante pour traiter les problèmes d'optimisation de grande taille si l'optimalité n'est pas primordiale. Elles permettent d'obtenir des solutions de qualité intéressante en un temps de calcul réduit.

Parmi ces méthodes approchées, les heuristiques sont des méthodes de résolution spécifiques à un problème dont la généralisation n'est pas aisée. Les méta-heuristiques permettent, quant à elles, de positionner la méthode sur un niveau plus élevé d'abstraction. Ainsi, leur transposition sera plus aisée. Elles pourront s'appliquer à plusieurs problèmes d'optimisation (BOUSSAÏD et collab. [2013], BLUM et ROLI [2003], TALBI [2009],

GENDREAU et POTVIN [2010]). Le but des méta-heuristiques sera de proposer des méthodes assurant un parcours efficace de l'espace de recherche. Ce parcours sera réalisé lors de la phase d'exploration de l'espace (diversification) consistant à parcourir différentes zones intéressantes de l'espace de recherche. Il sera aussi couplé à des phases d'exploitation (intensification) qui tenteront de dénicher les meilleures solutions de certaines zones. La principale différence entre les méthodes présentées réside dans le dosage de ces deux phases pendant l'optimisation.

Puisque nous proposerons une approche à base de méta-heuristiques et en particulier à base de recherche locale, nous présentons ci-après quelques concepts et algorithmes.

3.1.4 Principaux concepts des méta-heuristiques

Les quatre piliers des méta-heuristiques sont constitués par : la représentation d'une solution ou son encodage, la satisfaction des contraintes, la fonction objectif et l'analyse de performance.

Dans cette partie, nous traiterons rapidement ces quelques points qui seront indispensables à la bonne compréhension de nos choix et positionnements.

Représentation d'une solution Afin d'explorer le plus pertinemment possible l'espace de solutions et ainsi identifier les valeurs optimales, un codage d'une solution est indispensable. En général, ce codage apparaît sous la forme d'un vecteur de données. Ces données peuvent être binaires ou réelles. Cependant, il est aussi possible de trouver des solutions représentées sous la forme de graphes, ou de combinaisons de permutations. Avant de valider un codage, il est fondamental de respecter un principe de connexion. Ce principe, s'il est respecté, nous assure d'être en capacité d'identifier le chemin parcouru entre une solution et une autre dans un espace de recherche.

Satisfaction des contraintes Résoudre un problème d'optimisation combinatoire intègre naturellement l'apparition du respect de contraintes. Ainsi, chaque solution doit respecter les contraintes identifiées. Cependant, pendant la phase d'optimisation et de recherche de solutions correctes, il est possible de traiter différemment cette règle.

Les méthodes sont parfois amenées à rejeter immédiatement toute solution ne répondant pas à l'ensemble des contraintes. D'autres conservent la solution mais pénalisent la fonction objectif afin de reculer la solution dans le classement des solutions proposées. D'autres encore exploitent des solutions non faisables afin d'en générer d'autres. Il n'y a pas de recette absolue concernant la rigidité de l'intégration de ces contraintes. Notons cependant, qu'en première intention, les contraintes sont appliquées strictement.

Fonction objectif La définition de la fonction objectif revêt une importance majeure. En effet, il est fondamental d'identifier l'indicateur d'optimisation afin de nous assurer que l'on réponde à la bonne question. Cette définition n'est pas forcément naturelle ou aisée. Dans l'étude que nous avons menée, nous développerons ce point de manière spécifique.

La construction de la fonction objectif est parfois rendue difficile car il est en général souhaitable d'optimiser plusieurs indicateurs en même temps. Il peut alors s'agir d'optimisation multi-objectif. Une agrégation des différents objectifs peut aussi être envisagée (équation linéaire par exemple).

La fonction objectif guidera la méthode dans la découverte de l'espace de recherche. Elle constitue donc la pierre angulaire de la méthode.

Analyse de performance Comme indiqué précédemment, les méta-heuristiques ne permettent pas d'identifier l'optimum global. Ainsi, il est difficile de se prononcer sur la performance d'une méthode. C'est pourquoi, pour un problème d'optimisation combinatoire, les méthodes sont comparées. Ces comparaisons portent en général sur le temps de calcul, sur la qualité de la solution proposée... Dans le cas où la solution optimale à un problème est connue, il est pertinent de comparer la solution proposée à la solution optimale.

3.1.5 Méthodes de recherche locale

Sur la figure 3.1, nous avons présenté les deux grandes familles de méta-heuristiques :

- Méthodes à base de voisinage ou méthode à solution unique.
- Méthodes à base de population de solutions.

Les méthodes à base de voisinage, également appelées recherches locales, cherchent à parcourir l'espace de recherche en passant de solution en solution, de proche en proche. Les méthodes à base de population exploitent un ensemble de solutions et font évoluer la population de solutions en mettant en place des stratégies de sélection, de croisement et de mutation des solutions.

Nous présentons, pour notre part, uniquement les méthodes de recherches locales exploitées dans cette étude :

- Méthode de descente,
- Recherche Tabou,
- Méthode de recherche locale itérée.

Avant cela, nous définissons la notion de voisinage nécessaire aux recherches locales.

Notion de voisinage A toute solution s , il est possible d'associer un ensemble de voisins. Cet ensemble est généré à partir de s en appliquant de petites modifications sur s . Cette méthode nous permet d'explorer, sans changer radicalement la solution s , des solutions proches afin d'en évaluer la qualité. Ce petit changement sur s est opéré par un *opérateur de voisinage*.

Ainsi, l'on obtient la définition suivante : $\mathcal{N}(s)$, voisinage de s , est l'ensemble des voisins de s obtenu par l'application de l'opérateur de voisinage \mathcal{N} sur s .

Il est également possible de définir le terme d'optimum local par rapport à l'opérateur de voisinage $\mathcal{N}(s)$ dans le cas d'une minimisation par : s est un *optimum local* si et seulement si il n'existe dans le voisinage de s , aucune solution meilleure que s :

$$\forall s' \in \mathcal{N}(s), f(s) \leq f(s') \quad (3.2)$$

Méthode de descente Les méthodes de descente ou *Hill climbing (HC)* sont les recherches locales les plus simples. Partant d'une solution, elles recherchent dans le voisinage de celle-ci, une nouvelle solution permettant d'améliorer la fonction objectif. L'algorithme 3.1 décrit cette méthode pour un problème de minimisation.

Cette méthode atteindra donc naturellement un optimum local.

Le principal problème de la méthode de descente est de terminer la recherche lorsque tous les voisins possibles (issus d'un opérateur) ont été évalués et qu'un optimum local a été décelé. Il est possible d'avoir manqué une meilleure solution dans une autre zone de l'espace de recherche.

Algorithm 3.1 Simple Hill Climbing Algorithm

```

Given  $s \in \mathcal{D}$ 
while Stopping criterion not met do
  Select  $s' \in \mathcal{N}(s)$ 
  if  $f(s') < f(s)$  then
     $s \leftarrow s'$ 
  end if
end while

```

La mise en œuvre d'une telle méthode, nécessite de préciser le type d'exploration de voisinage à utiliser. En effet, il existe différentes méthodes de sélection du voisin à considérer.

La stratégie du *First Improvement* par exemple sélectionne parmi les solutions voisines la première solution qui améliore la fonction objectif. Cette méthode n'impose pas naturellement la définition d'un critère d'arrêt puisque la descente se poursuit jusqu'à ce qu'il n'y ait plus de voisin permettant d'améliorer la solution courante. Il est cependant possible d'y ajouter des critères d'arrêt sur le temps de calcul, sur le nombre d'itérations ou tout autre indicateur concernant par exemple la qualité de la solution proposée.

La stratégie du *Best improvement* est parfois utilisée. Elle permet, à chaque itération, parmi un voisinage, de sélectionner le meilleur voisin. Cette stratégie impose donc une évaluation de tous les voisins d'une solution au risque d'allonger le temps de calcul. En effet, le recours à la fonction d'évaluation peut être très consommateur de ressources.

Recherche Tabou La recherche Tabou introduite par F. Glover (GLOVER [1989], GLOVER et LAGUNA [2013]) ajoute une mémoire à court terme à l'algorithme.

L'algorithme accepte parfois des solutions qui n'améliorent pas toujours la solution courante. L'algorithme met en œuvre une liste taboue (tabu list) T de longueur k contenant les k dernières solutions visitées. Cette liste ne donne pas la possibilité à une solution déjà trouvée d'être acceptée et stockée dans la liste tabou. Alors le choix de la prochaine solution est effectué sur un ensemble des solutions voisines en dehors des éléments de cette liste taboue. Quand le nombre k est atteint, chaque nouvelle solution sélectionnée remplace la plus ancienne dans la liste. La construction de la liste taboue est basée sur le principe FIFO, c'est-à-dire le premier entré est le premier sorti.

Comme critère d'arrêt, on peut par exemple fixer un nombre maximum d'itérations sans amélioration de s^* , ou bien fixer un temps limite à la suite duquel la recherche doit s'arrêter.

La taille de la liste Tabou peut être pilotée dynamiquement (en cours de calcul) mais est souvent définie a priori, en amont de la résolution.

Le principe de cette méthode est présenté ci-dessous :

Méthode de recherche locale itérée La recherche locale itérée ou *Iterated local search* (ILS) exploite une nouvelle fois le principe de la descente (LOURENÇO et collab. [2010]). Cependant, lorsque la méthode atteint un optimum local, une perturbation forte est appliquée à la solution par un opérateur de perturbation. Le choix de l'opérateur est très important car il doit permettre de sortir de l'optimum local et de s'en éloigner suffisamment pour ne pas y revenir.

La figure 3.2 permet de présenter l'opération.

Algorithm 3.2 Tabu Search

```
Given  $s \in \mathcal{D}$ 
 $s^* \leftarrow s$ 
Clear T
while Stopping criterion not met do
  Select the best neighbor  $s' \in \mathcal{N}(s)$  and  $s' \notin T$ 
   $s \leftarrow s'$ 
  Update T
  if  $f(s) < f(s^*)$  then
     $s^* \leftarrow s$ 
  end if
end while
```

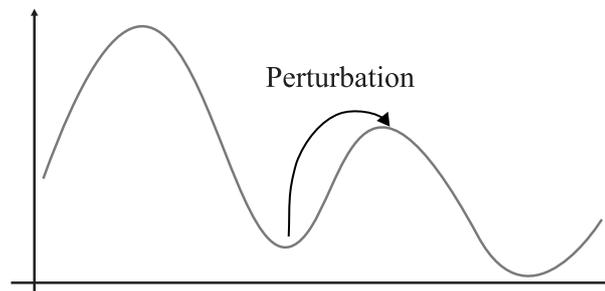


FIGURE 3.2 – Perturbation lors d'une recherche locale itérée

Une fois le nouveau voisin généré, une nouvelle descente est exécutée espérant ainsi explorer une nouvelle partie de l'espace de recherche et trouver une meilleure solution.

Cette méthode impose un critère d'arrêt et demande un paramétrage pertinent pour l'opérateur de perturbation.

3.2 Méthode d'optimisation proposée

Ayant très brièvement introduit quelques concepts fondamentaux, nous allons dans un premier temps présenter en quoi notre problème d'optimisation peut être considéré comme « original ». Ensuite, nous définirons les caractéristiques de notre problème à savoir : une solution, un opérateur de voisinage de solution et détaillerons les techniques déployées pour identifier la meilleure solution possible. Enfin, nous présenterons quelques essais réalisés permettant de valider des paramétrages et d'apprécier la performance de notre approche.

3.2.1 Recherche d'un bon ensemble de décisions : un problème d'optimisation combinatoire ?

Dans le chapitre précédent, nous avons présenté ce qu'est une solution pour notre problème et comment la construire. Nous avons aussi pu remarquer que nous nous trouvons face à un problème combinatoire car de nombreuses solutions sont envisageables. De plus, il nous est impossible de toutes les énumérer.

Nous nous trouvons également confrontés à un problème pour lequel il n'y a pas de fonction objectif globale. Ainsi, associer une qualité à une solution n'est pas possible dans notre contexte de jeu. Ceci en fait une spécificité importante. En effet, dans le contexte du

xbeer game (NACCACHE et collab. [2010]), une évaluation de chaque solution, via un score est réalisable, rendant une modélisation en un problème d'optimisation possible.

Dans notre cas, cette évaluation n'existe pas. Cependant, il est possible de mettre deux solutions en concurrence et de les comparer pour déterminer quelle est la meilleure des deux. Nous allons donc exploiter cette caractéristique et proposer une méthode à base de voisinage.

Nous déciderons qu'une solution est meilleure qu'une autre si elle obtient un meilleur score. La construction du score sera identique à la méthode implantée dans le simulateur.

En raison des deux arguments précédemment cités, une approche par recherche locale est envisageable.

3.2.2 Représentation d'une solution

La représentation d'une solution doit contenir toutes les décisions possibles. Par conséquent, elle consiste en un vecteur d'entiers représentant par exemple, le *prix*, la *publicité*, les *options offertes*, les efforts en *R&D* et les *conditions de paiement* de chaque produit, sur chaque marché, sur chaque tour considéré pour l'optimisation. Le vecteur solution n'est composé que d'entiers. Ces entiers représentent des modalités de décisions comme l'indique la figure 3.3.

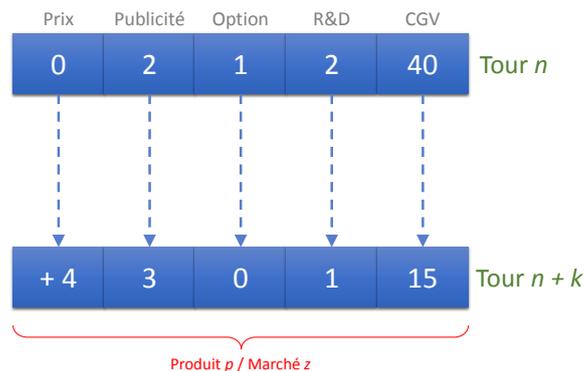


FIGURE 3.3 – Représentation du vecteur solution pour un produit p sur un marché z

Afin de clarifier cet aspect, nous pouvons citer deux exemples. Notre premier exemple correspond au prix de vente défini pour un couple produit/marché. Sur le vecteur solution, celui-ci sera compris entre 0 et 9. Chaque modalité correspondra, dans le simulateur, à une modulation d'un prix de marché habituellement pratiqué.

Par exemple, la modalité 5 correspond au prix de marché (indice 100), la modalité 4 à un prix inférieur à 5% (indice 95) du prix de marché et la modalité 7 à un prix supérieur de 12% au prix de marché. Ces indices de définition de prix par rapport au prix de marché sont définis arbitrairement dans le paramétrage de la simulation. Le prix de marché est également fixé dans la simulation par le scénario de marché. Il peut évoluer au cours des cycles. La représentation du vecteur, la valeur réelle du prix converti en euros n'aura aucun impact sur la génération de solution. L'impact sera ressenti dans le programme linéaire du département finance car il impacte la marge du produit. Cet indicateur impactera aussi la loi de marché. Nous reviendrons sur cet aspect dans la section suivante.

Concernant la dimension publicité, les modalités correspondent au numéro des actions de publicité lancées. A chaque numéro d'actions, un impact commercial sur le mar-

ché est positionné. Chaque action entraîne également un coût direct exprimé en euros. Pour cette modalité, il n'y a aucune relation directe réalisable facilement entre le vecteur solution et la décision opérationnelle. Il ne s'agit que d'une référence.

Ainsi, deux hypothèses différentes correspondent à deux solutions différentes. L'espace de recherche est donc composé de l'ensemble des hypothèses possibles pour tous les couples produits/marché et pour les tours prospectifs envisagés. Nous exposons dans les prochains paragraphes nos choix concernant les modes de génération des voisins.

3.2.3 Un voisinage variable

Afin de mettre en œuvre la recherche locale, nous définissons un opérateur de voisinage. Cet opérateur se construit en plusieurs étapes. Nous avons représenté sur la figure 3.4 les différentes étapes suivies qui permettent de passer d'une solution initiale à un voisin.

Dans notre représentation, la solution initiale comprend cinq variables (*le prix, la publicité, les options, la R&D et le délai de paiement client* indiqué sous l'intitulé *CGV*). Ces variables vont évoluer en fonction d'un vecteur, appelé *pas*, paramétré dans la méthode. Ce pas est constitué d'une valeur par variable de la solution s . Par tirage aléatoire de la variable impactée et du sens d'application du vecteur sur la solution s apparaît le voisin de s .

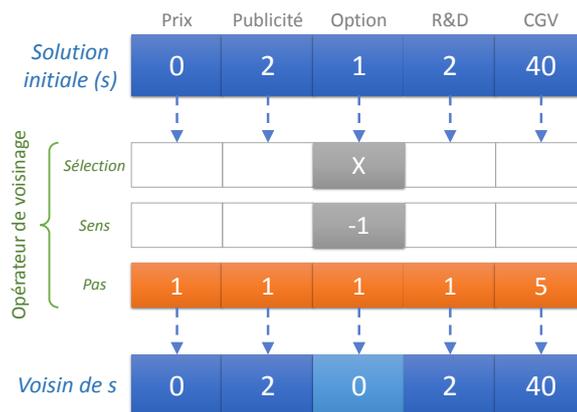


FIGURE 3.4 – Mode de génération des voisins

L'opérateur de voisinage est alors constitué :

- d'une sélection de variable (sélection aléatoire),
- d'un sens d'application (sélection aléatoire),
- d'un pas (paramétrage).

Le pas peut ainsi être adapté pendant l'exploration de différents espaces de recherche. Cette adaptation peut être réalisée a priori ou en cours d'optimisation. Le pas peut également évoluer pour chaque couple de produit/marché. Nous utilisons une méthode de parcours aléatoire des voisins et optons pour une méthode de sélection du voisin en suivant la stratégie du *First Improve*.

Pour explorer efficacement l'espace de recherche et donc envisager un plus grand nombre de solutions variées, plusieurs paramétrages de voisinage ont été testés.

La génération de voisins est similaire pour l'ensemble de l'hypothèse composée par tous les produits sur tous les marchés pour les k tours pris en compte dans la représentation de la solution.

Afin d'accélérer l'exploration de l'espace de recherche, nous proposons plusieurs étapes pour le parcours de celui-ci. Nous définissons dans un premier temps des pas à la « granulométrie » variée à savoir :

- un **grand** pas : vecteur composé de valeur importante (3 modalités) permettant la création d'un voisin assez éloigné de la solution initiale,
- un pas **moyen** : vecteur composé de valeur médiane (2 modalités) permettant la création d'un voisin assez proche de la solution initiale,
- un **petit** pas : vecteur composé de valeur faible (une modalité) permettant la création d'un voisin très proche de la solution initiale.

Nous appliquerons donc en première approche un grand pas. Celui-ci nous permettra de parcourir rapidement l'espace de recherche. Dès l'atteinte d'un optimum local, nous appliquerons un pas moyen ou petit qui nous permettra d'affiner notre parcours.

Par exemple, si l'on considère la variable *Délai de paiement* (CGV), la solution initiale ou courante pourrait être de 45 jours. Dans la première approche, un pas de 10 jours permettrait d'obtenir l'ensemble suivant de valeurs possibles : {15; 25; 35; 45; 55; 65; 75}. Supposons que, après la première optimisation, l'algorithme converge sur la valeur de 25 jours. Nous souhaitons affiner la recherche et décidons d'appliquer un nouveau pas de 5 jours. Cette modification nous permettra d'obtenir l'ensemble de valeurs possibles suivant : {15; 20; 25; 30; 35; ...}.

3.2.4 Une procédure itérative

Afin d'échapper au minimum local atteint en fin d'application de descente, nous avons mis en œuvre un opérateur de perturbation et un schéma itératif conformément à la méthode *Iterated Local Search* précédemment exposée.

L'opérateur de perturbation est défini simplement par l'appel successif de l'opérateur de voisinage. Nous choisissons d'appliquer à la solution l'opérateur de pas le plus grand. Celui-ci nous éloignera le plus possible de l'optimum local trouvé. De plus, afin de ne pas évaluer des solutions déjà visitées en début d'optimisation, nous avons implémenté une liste des solutions déjà visitées et donc interdites. Ce stockage de solutions est une liste taboue. Différentes longueurs de listes ont été expérimentées sans qu'aucun impact n'ait été décelé.

Le critère d'arrêt, exposé ultérieurement, intervient par contre dans la performance.

Le principe de perturbation retenu nous impose d'accepter systématiquement la nouvelle hypothèse générée après la perturbation. Aucun critère de qualité n'entre en considération même si celle-ci n'est pas compétitive par rapport à la solution à partir de laquelle elle a été générée.

Avec ce type de procédure itérative, nous devons définir un critère de fin. Nous avons opté pour le nombre maximal d'itérations. Ce choix nous permettra de comparer équitablement plusieurs paramétrages de la méthode. Le nombre limite a été fixé après quelques expérimentations afin de respecter un temps de calcul/résolution acceptable.

3.2.5 Évaluation d'une solution

L'évaluation d'une solution se fait par comparaison par rapport aux autres solutions. La figure 3.5 présente le tournoi que nous avons mis en place qui nous permet de sélectionner la meilleure solution parmi les solutions testées du voisinage parcouru.

Chaque voisin représenté de 1 à h est issu de la solution courante. Ces voisins sont utilisés comme hypothèses et sont injectés au processus d'optimisation interne. Cette

optimisation correspond à la construction d'une solution par l'optimisation des départements Production, Distribution et Finance comme présenté dans le chapitre 2. Pour mémoire, chaque voisin n'aboutit pas systématiquement à une solution. En effet, la construction d'une solution peut échouer pour diverses raisons : insuffisance de moyens financiers, temps d'optimisation trop long, capacité de distribution trop faible...

Au terme du tournoi d'évaluation, la meilleure solution issue du voisin i est sélectionnée. Ce voisin permettra à son tour de générer un nouvel ensemble de voisins par application de l'opérateur de voisinage.

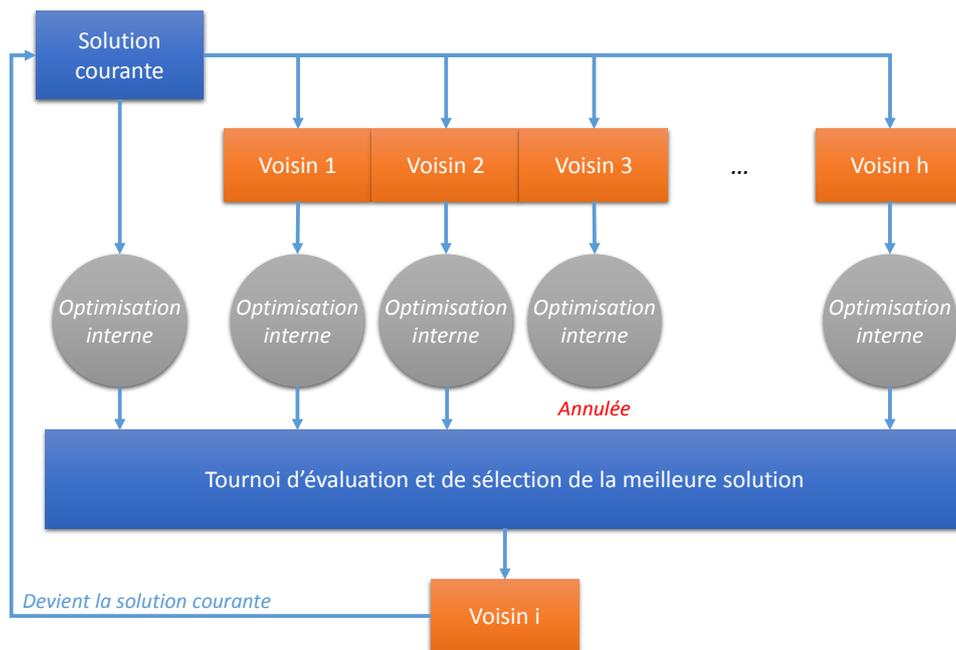


FIGURE 3.5 – Mode de sélection de solution

Nous devons être vigilants sur le nombre d'évaluations à réaliser car l'évaluation est coûteuse. En effet, la phase d'optimisation interne peut être longue. La confrontation des équipes en tournoi l'est. Elle impose la mise en œuvre de la confrontation concurrentielle complète qui entraîne avec elle de nombreux traitements et accès aux données conjoncturelles du jeu.

3.3 Expérimentations

Les expérimentations que nous présentons dans les paragraphes suivants ont pour objectifs :

- de valider le design de l'approche globale de la méthode,
- d'établir les paramétrages les plus adéquats nous offrant les solutions les plus performantes par rapport aux décisions prises par des joueurs réels,
- de valider la performance globale de la méthode.

3.3.1 Protocole expérimental

Jeux de données

Afin de valider la méthode d'optimisation, nous avons déployé nos expérimentations sur des parties déjà jouées pendant des animations pédagogiques classiques en contexte normal. Les étudiants provenaient d'écoles de management et d'écoles d'ingénieurs et disposaient des pré requis nécessaires à l'appréhension des notions de base en finance d'entreprise.

Nous avons sélectionné des simulations sur deux scénarios économiques différents :

- *Market₁* : marché très concurrentiel présentant de petits volumes de ventes. Huit équipes en compétition. Ce marché impose que les équipes répondent au mieux aux composantes du marketing mix sur chaque segment de marché. L'échec ou le succès d'une équipe peut se décider sur un nombre très limité de décisions.
- *Market₂* : six équipes en compétition sur un marché de produits de consommation. Les volumes de ventes évoluent très rapidement. Les entreprises doivent être capables de produire de grosses quantités de produits. Elles doivent également réaliser de gros efforts en marketing et communication afin d'atteindre une position dominante sur le marché.

Ces deux marchés sont très différents et offrent un bon terrain d'expérimentation. Ils nous permettront d'apprécier les performances des deux approches. Le but de ces expérimentations est de définir les paramètres les plus performants pour la méthode d'exploration choisie.

Méthodologie de la comparaison

Les expérimentations menées nous permettent d'analyser les points suivants :

- Stratégie de voisinage : Paramétrages du voisinage
- Recherche locale itérative (ILS) : Analyse de l'opérateur de perturbation
- Temps de calcul : Analyse du nombre maximal d'itérations

Ainsi, pour évaluer l'efficacité des différentes configurations proposées, nous exploiterons le même ensemble d'équipes (même jeu concurrentiel). Par conséquent, nous remplacerons une équipe de joueurs réels par la méthode d'optimisation. L'équipe remplacée sera toujours la même pour toutes les expérimentations. Nous comparerons sa performance aux autres équipes en lice.

Évaluation des performances

Concernant la première série d'expériences, qui a pour objet de déterminer le meilleur paramétrage, tous les tests effectués ont été lancés douze fois par cas puisque la méthode est stochastique.

Concernant la deuxième série d'expériences, qui a pour objet d'évaluer l'efficacité globale de l'optimisation, nous prendrons la place de toutes les équipes inscrites au jeu successivement et pour les deux sessions.

Le critère de performance que nous avons choisi est le rang/classement obtenu. Ce rang est défini dans la section 1.1.6.

Notons que nous exploitons la loi de marché intégrée au simulateur UpStrat avec les données connues des concurrents (puisque nous remplaçons une équipe parmi les inscrits). Nous gommons ainsi la non connaissance du marché et de sa réaction. Cette si-

tuation ne se retrouve pas en situation réelle, puisque lorsqu'une équipe prend ses décisions, elle ignore les décisions des autres équipes et la réaction du marché lors de la confrontation de ces décisions. En figeant ainsi les éléments exogènes à notre processus d'optimisation, nous observons seule la performance de notre algorithme de recherche locale.

Dans le chapitre suivant, nous évaluerons notre proposition d'approche permettant d'anticiper les décisions des concurrents et ainsi de se mettre en situation réelle.

3.3.2 Comparaison des opérateurs de voisinage

La notation suivante (*Prix/Publicite/R&D/Option/Delai_de_paiement*) définit l'opérateur de voisinage.

Par exemple, le vecteur 1/1/1/1/5 autorise l'opérateur de voisinage à modifier le prix, la publicité, la R & D et les options d'une seule modalité. Par contre, le délai de paiement peut varier de 5 unités (ici le jour pour ce critère).

Afin de déterminer l'impact de ces opérateurs, nous avons défini 3 opérateurs de voisinage comme suit :

- **Pas 1** : opérateur 1/1/1/1/5 noté *thin*
- **Pas 2** : opérateur 2/2/2/2/10 noté *medium*
- **Pas 3** : opérateur 3/3/3/3/15 noté *large*

Le tableau 3.1 présente les résultats obtenus après avoir implémenté plusieurs combinaisons d'opérateurs.

TABLEAU 3.1 – Stratégie de voisinage - Statistiques sur douze exécutions

Scénario	Market ₁			Market ₂		
	A1	B1	C1	D1	E1	F1
Test ID	1	2+1	3+2+1	1	2+1	3+2+1
Nombre de pas	1	2+1	3+2+1	1	2+1	3+2+1
Moyenne	3,4	1,8	1,8	2,6	2,1	2,0
Q1	3	1	1	2	1,75	1,75
Min	2	1	1	1	1	1
Max	4	3	3	5	3	4
Q3	4	2	3	3	3	2

Dans ce tableau, les tests A, B, C ont été effectués avec le marché 1 et les tests D, E et F avec le marché 2. Les tests A1 et D1 utilisent le pas 1 seul. Les tests B1 et E1 exploitent le pas puis le pas 1. Enfin, les tests C1 et F1 utilisent les trois pas en commençant par le pas 3 puis 2 puis 1.

La génération du voisinage exploite le pas $n - 1$ quand aucune amélioration de résultat n'a été trouvée pour le pas n . Ceci signifie qu'un pas plus petit est exploité lorsqu'un optimum local pour un opérateur de voisinage a été identifié.

Nous n'avons pas exploité le schéma itératif pour ces expérimentations afin de nous focaliser sur la performance des opérateurs et leur stratégie d'exploitation. Ainsi, la méthode est stoppée lorsqu'un optimum local est atteint.

Nous avons exploité le rang comme indicateur de performance. Pour rappel, le rang d'une solution est issu d'un classement pondéré de six critères financiers et d'activité comme présenté au chapitre 1. Ainsi, plus le rang obtenu est faible meilleure est la performance de la solution.

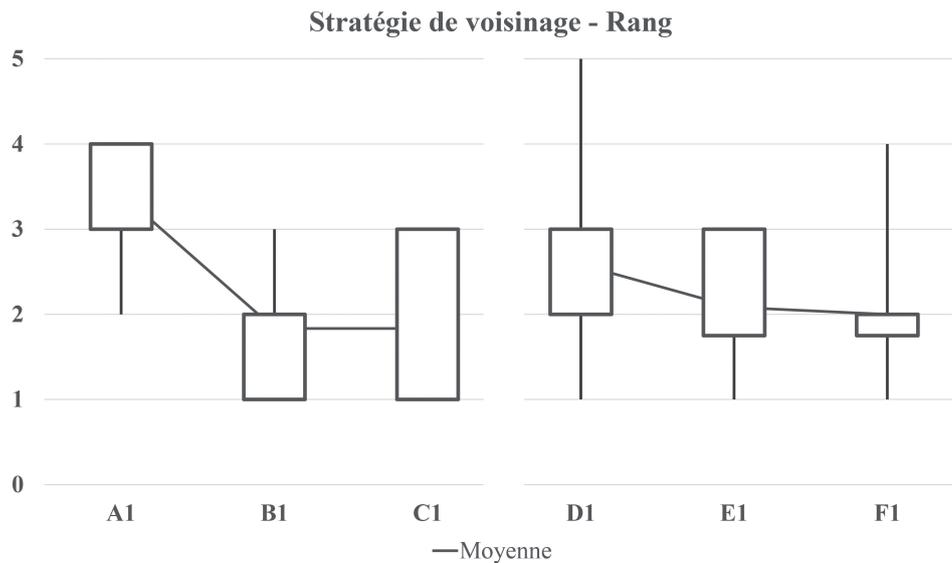


FIGURE 3.6 – Box plot de la stratégie de voisinage

La figure 3.6 offre une visualisation des résultats obtenus. Pour chaque série de tests, le box plot obtenu nous permet de voir la variation des rangs obtenus sur les différentes exécutions. La ligne entre chaque box plot relie les moyennes de chaque expérimentation.

En appliquant les différentes phases présentées, le tableau 3.1 et la figure 3.6 nous permettent de conclure que l'approche proposée nous positionne de la première à la quatrième place sur huit équipes pour le $Market_1$ et de la première à la cinquième place sur six équipes pour le $Market_2$. De plus, les résultats A1 et D1 nous indiquent que l'application d'un seul pas fin n'est pas suffisant. L'exploration de l'espace de recherche s'est arrêtée trop prématurément et la qualité de la solution retenue n'est pas satisfaisante.

Les combinaisons des différentes phases nous permettent d'accroître significativement la qualité de la solution au détriment du temps de calcul. Les tests A1 et D1 qui ne mettent en œuvre qu'un seul pas sont réalisés en 90 secondes. Ce temps de calcul atteint 180 secondes pour les tests C1 et F1.

3.3.3 Analyse de l'opérateur de perturbation

Le même protocole expérimental est utilisé afin d'examiner les opérateurs de perturbation. Pour l'opérateur de voisinage, le pas 2 ajouté au pas 1 sont choisis. En effet, ce choix se justifie par un bon compromis entre temps de calcul et performance.

Nous prenons le pas 3 comme opérateur de perturbation. Ce pas large nous permet de nous éloigner fortement de la solution initiale. Il est appliqué plusieurs fois. Dans ces expériences, nous avons choisi de modifier le nombre d'applications de cet opérateur de perturbation.

Les applications comparées sont les suivantes :

- A2, D2 : 1 application du pas 3 de l'opérateur de voisinage
- B2, E2 : 2 applications du pas 3 de l'opérateur de voisinage
- C2, F2 : 3 applications du pas 3 de l'opérateur de voisinage

Le tableau 3.2 et la figure 3.7 présentent les résultats en terme de rang.

Ces résultats montrent qu'il semble préférable d'utiliser un opérateur de perturbation aussi large que possible. En effet, la qualité des solutions s'améliore régulièrement

TABLEAU 3.2 – Opérateur de perturbation - Statistiques sur 12 exécutions

Scénario	<i>Market₁</i>			<i>Market₂</i>		
Test ID	A2	B2	C2	D2	E2	F2
# times	1	2	3	1	2	3
Moyenne	2,0	1,8	1,6	2,5	2,4	2,0
Q1	2	1	1	2	1	2
Min	1	1	1	1	1	1
Max	3	3	3	6	5	3
Q3	2	2	2	3	4	2

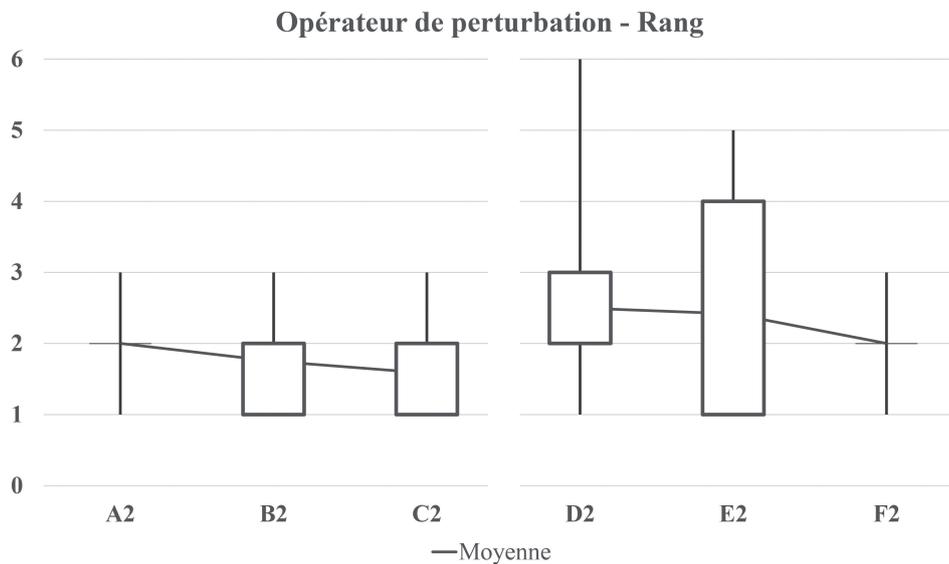


FIGURE 3.7 – Box plot de l'opérateur de perturbation

lorsque le nombre de répétitions de l'opérateur augmente. Cependant, les règles du jeu nous empêchent d'augmenter indéfiniment le nombre d'applications de l'opérateur de perturbation.

Si l'on se concentre sur la variable *prix*, celle-ci varie entre les modalités -4 à 4. Nous ne pouvons nous écarter de cette fourchette. Il est donc inutile d'appliquer à de multiples reprises l'opérateur de perturbation au risque d'être limité par les bornes des variables et ainsi de rester sur des espaces de recherche déjà parcourus.

3.3.4 Impact du nombre total d'évaluations

Le protocole expérimental précédent était basé sur un temps de calcul maximal. Nous avons opté pour un temps de calcul « raisonnable » d'environ deux minutes pour les plus courtes et de trois minutes pour les évaluations les plus longues.

Pour ce délai, le nombre maximal d'évaluations possibles varie entre 100 à 200. Dans ce paragraphe, les expériences comparent les résultats obtenus après 100, 150 et 200 évaluations afin d'analyser l'impact du critère d'arrêt.

Tous les résultats présentés sont générés avec le même protocole expérimental.

TABLEAU 3.3 – Nombre d'évaluations - Statistiques sur 12 exécutions

Scénario	Market ₁			Market ₂		
	A3	B3	C3	D3	E3	F3
Test ID	A3	B3	C3	D3	E3	F3
# Eval.	100	150	200	100	150	200
Moyenne	2,58	1,50	1,42	2,17	1,92	1,83
Q1	2	1	1	1	2	2
Min	1	1	1	1	1	1
Max	4	3	3	8	3	3
Q3	3	2	2	2	2	2

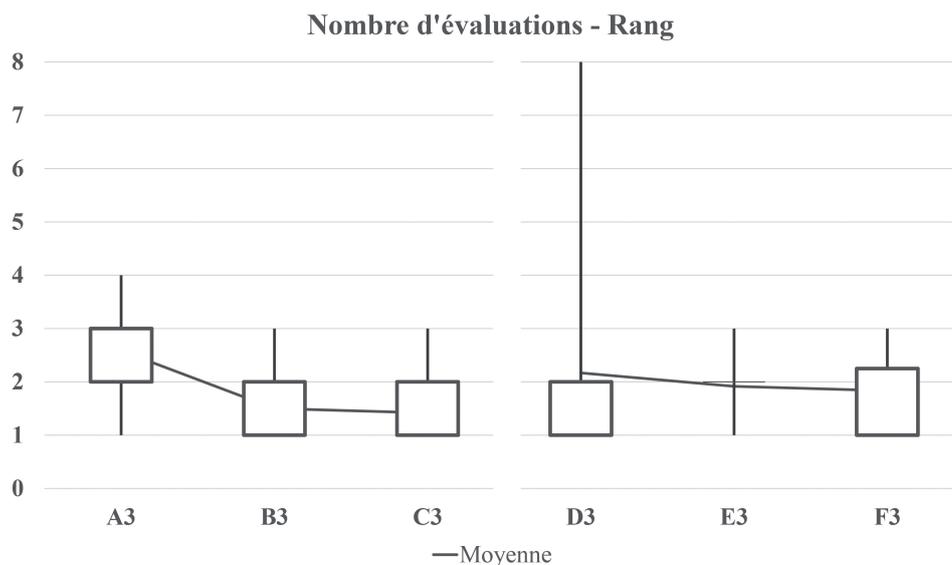


FIGURE 3.8 – Rang obtenu par variation du nombre maximal d'évaluations

Le tableau 3.3 et la figure 3.8 montrent que le nombre maximal d'évaluations impacte la qualité de la solution. Cependant, nous remarquons qu'au-delà de 150 évaluations, le processus ne permet plus d'améliorer le résultat/classement de manière significative.

Nous obtenons une légère amélioration qui entraîne cependant une extension du temps de calcul. Le faible gain en performance devient très coûteux.

Ces observations nous permettent de dire que l'espace de recherche est très accidenté. Sur un point de vue stratégique ou financier, cela peut sembler cohérent puisqu'il n'existe pas de stratégie gagnante a priori. Il y aurait donc un ensemble de stratégies/groupes de solutions performantes dans le contexte économique donné.

Ce contexte économique englobe les données de marché comme la croissance, la demande et les facteurs clés de décision qui régissent les clients. Mais, il s'accompagne aussi du contexte concurrentiel animé par les participants eux-mêmes. Chaque jeu est différent.

Notons également qu'il n'est pas aisé de fixer le nombre maximal d'évaluations. En effet, la taille du voisinage n'est pas fixe. En début de simulation, pour les premiers tours de jeu (1 et 2 par exemple), les options offertes via les efforts en R&D ne sont pas accessibles, tous les marchés ne sont pas ouverts, les entreprises ne disposent pas toujours de leur outil de production... Par la suite, l'ensemble des choix et options est disponible et le nombre de voisins augmente considérablement.

Ces expérimentations confirment qu'après un petit nombre d'itérations, la méthode converge vers de très bonnes solutions. L'algorithme atteint régulièrement le premier rang.

Par conséquent, afin de préserver un calcul rapide mais aussi une qualité de solution correcte, nous avons réalisé un compromis et fixé à 150 évaluations.

Les différents paramètres de la méthode ayant été fixés (tableau 3.4), il est maintenant possible d'en évaluer la performance globale.

TABLEAU 3.4 – Paramètres retenus pour la recherche locale

Paramètre	Valeur
Opérateurs de voisinage	Pas 2 + Pas 1
Nombre d'application de l'opérateur de perturbation	3 fois le pas 3
Nombre d'évaluations	150 évaluations

3.3.5 Performance de la méthode

Afin d'examiner l'efficacité globale du processus mis en œuvre, nous avons sélectionné un ensemble de parties déjà jouées. Dans ces parties, il y avait au minimum cinq équipes en lice. Nous avons ainsi remplacé successivement toutes les équipes par le processus d'optimisation en archivant le classement occupé par l'équipe remplacée dans la partie pour le tour considéré. Ainsi, lorsque nous avons remplacé l'équipe arrivée en première position, le processus était alors en concurrence avec les équipes classées de la deuxième à la énième place. Notre objectif est de voir si la solution proposée par la méthode parvient à se classer en meilleure position que l'équipe réelle remplacée.

Instinctivement, nous pensons que le processus d'optimisation ne rencontrera pas de difficulté particulière pour se classer en tête lors du remplacement de l'équipe réelle qui était déjà en tête. Cependant, qu'en est-il pour les équipes en fin de classement et donc, qu'en est-il si l'on confronte la méthode aux meilleurs concurrents?

La figure 3.9 présente les résultats du remplacement des équipes. Pour chaque exécution, nous avons indiqué le classement final de la solution produite par la méthode à l'aide de box plot pour étudier la robustesse de l'approche. Cette figure montre que, quelle que soit l'équipe qui est remplacée, la méthode est capable d'atteindre le premier

rang. Elle obtient de très bons résultats et est assurément meilleure que les participants eux-mêmes.

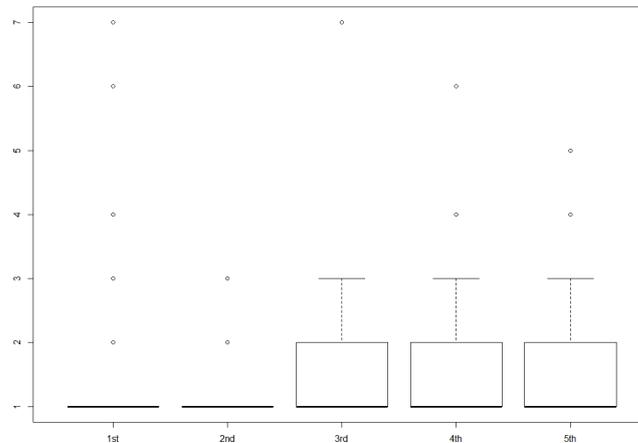


FIGURE 3.9 – Classements à l’issu des remplacements successifs

Nous avons également souhaité connaître le nombre d’évaluations à réaliser pour atteindre la meilleure place possible. Pour disposer de cette information, nous avons sélectionné une partie parmi des parties testées. Cette partie comportait cinq équipes et se déroulait sur le marché nommé $Market_1$. Ainsi, pour chaque évolution du classement et lors de l’application de l’opérateur de perturbation intégré à l’Iterative Local Search, nous avons relevé le nombre d’évaluations faites. Le tableau 3.5 présente les résultats de cette expérimentation.

TABLEAU 3.5 – Evaluation de la performance après remplacement des équipes

Équipe 1ère		Équipe 2ème		Équipe 3ème		Équipe 4ème		Équipe 5ème	
#Eval.	Rang								
5	4	5	5	5	3	5	3	5	4
9	4	9	5	9	2	9	3	9	3
17	3	21	4	13	1	13	3	21	3
21	2	33	3			21	3	49	3
97	2	45	3			25	3	53	3
101	2	121	3			61	3	73	3
113	1	129	2			65	1	77	2
		141	1					85	1

Notons aussi que le nombre d’itérations pour atteindre le résultat final est assez faible. Cet indicateur nous prouve l’efficacité de la méthode et nous assure un temps de calcul compatible avec le cadre d’utilisation prévu de la méthode.

3.4 Conclusions

Pour résumer ce chapitre, nous avons déployé une méthode de recherche locale afin d’identifier les solutions les plus performantes de notre espace de recherche. Ce choix avait été motivé par une contrainte de temps d’exécution résolument court et surtout

par la possibilité de comparer la performance d'une solution par rapport à d'autres sans pouvoir exprimer un critère de qualité absolu. Nous pouvions nous contenter de bonnes solutions. Ainsi l'approche par voisinage semblait totalement indiquée.

La recherche locale a aussi été hybridée avec une méthode exacte basée sur la programmation linéaire mixte. Nous nous sommes focalisés ici sur le dimensionnement de la recherche locale et en particulier sur les performances des paramètres de parcours de l'espace de recherche.

Les expériences réalisées montrent que les performances de la méthode dépendent des paramètres suivants :

- Opérateur de voisinage
- Opérateur de perturbation
- Nombre maximal d'évaluations

Une fois les paramètres choisis, nous avons comparé les performances de la méthode proposée aux performances obtenues par les équipes de joueurs. Il semble que la méthode permet d'obtenir des résultats aussi bons que les joueurs et bien souvent meilleurs. Nous pouvons donc valider cette approche.

Nous pensons que ces paramètres pourraient être fixés dynamiquement pendant les phases de jeu. En effet, la taille du voisinage augmente significativement en cours de simulation et il semblerait pertinent d'adapter les paramètres de découverte du voisinage à ces modifications substantielles. Il serait également intéressant d'étudier le lien entre ces paramètres et l'intensité concurrentielle : nombre de concurrents, comportement de ces derniers. Ces valeurs seraient aussi sujettes à évolution en fonction des modèles de marché sélectionnés par le pédagogue lors du lancement de sa session de formation.

Notons également que dans ce chapitre, nous n'avons pas intégré de modifications des comportements des joueurs concurrents par rapport aux résultats obtenus. En effet, tous les joueurs (autres que l'optimisation) étaient des joueurs qui ne pouvaient réagir en direct, puisque leur partie était déjà jouée. Dans la suite de ce document, nous nous intéresserons aux comportements des joueurs. Une meilleure connaissance des habitudes de ces derniers nous permettrait d'anticiper leurs réactions et décisions et nous pourrions adapter notre optimisation au contexte concurrentiel alors connu.

Chapitre 4

Analyse du comportement des joueurs pour simuler la loi de marché

Une caractéristique importante des business Games comme UpStrat est que toutes les équipes enregistrées sur un jeu, prennent leurs décisions simultanément.

Au sein du chapitre 3, nous avons occulté cette question et avons fourni les décisions des concurrents à la méthode avant même l'optimisation afin de permettre de simuler la réaction du marché, phase indispensable pour estimer les ventes et procéder ensuite aux différentes optimisations (voir figure 4.1). Ceci avait pour objectif de valider la méthode d'optimisation.

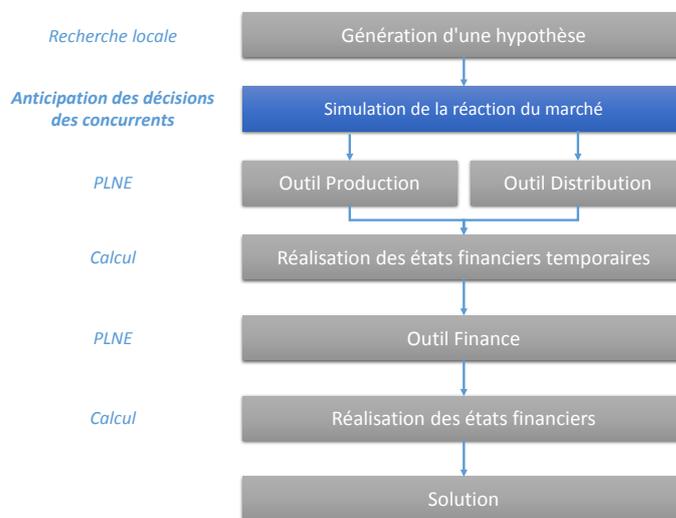


FIGURE 4.1 – Génération d'une solution

Cependant, en situation réelle, les décisions des concurrents ne peuvent être connues à l'avance et il est nécessaire de pouvoir les anticiper. C'est ce que nous proposons de faire dans ce chapitre.

A ce stade de l'étude, nous pourrions opter pour plusieurs méthodes de prédiction ou d'anticipation. Nous pouvons par exemple exploiter des méthodes basées sur *la théorie des jeux* ou encore des méthodes d'*extraction de connaissances*.

Notons que la théorie des jeux a largement été utilisée en stratégie économique, diplomatique, lors d'appels d'offres... Il est donc tout naturel d'imaginer que le contexte de cette étude soit favorable à l'exploitation de telles théories. Nous proposons donc, dans

un premier temps, de présenter la théorie des jeux et sa transposition sur les business games dans un second temps. Nous présenterons par la suite notre approche, basée sur des méthodes d'extraction de connaissances, et les appliquerons à notre cas d'étude.

4.1 Approche par la théorie des jeux

L'objectif de la théorie des jeux est de traiter des situations dans lesquelles le résultat d'une décision d'une personne ou d'un joueur dépend de sa décision mais aussi des décisions prises par les personnes avec qui il interagit (EASLEY et KLEINBERG [2010], NISAN et collab. [2007] et SHOHAM et LEYTON-BROWN [2008]). Cette théorie se retrouve dans de nombreux contextes : modélisation du trafic routier, modélisation de jeux de stratégie (échecs), de jeux de cartes... Les exemples cités précédemment imposent aux acteurs confrontés à ces situations de mettre en œuvre une stratégie. Ainsi, la théorie des jeux analyse ces stratégies et en évalue les performances.

4.1.1 Qu'est-ce qu'un jeu ?

En théorie des jeux, les ingrédients standards que l'on retrouve dans un jeu sont :

- Un ensemble de participants que l'on nomme les *joueurs*
- Chaque joueur a , à sa disposition, un ensemble d'options, de choix à réaliser. Ces options sont appelées *stratégies*.
- Pour chaque choix, chaque joueur reçoit un *gain* qui peut dépendre de la stratégie des autres joueurs. Ces gains sont souvent exprimés de manière numérique. Ils sont, dans des cas simples de jeu à deux joueurs, représentés sous la forme d'un tableau ou *matrice des gains*.

Dans la plupart des cas, l'objectif des joueurs est de maximiser leur gain à l'issue de leur choix. Il est important que toutes les stratégies envisageables pour chaque joueur soient connues de tous. Il est aussi important que la matrice des gains soit elle-même connue et partagée.

4.1.2 Le dilemme du prisonnier

Nous proposons de présenter une illustration typique en théorie des jeux. Le dilemme du prisonnier a fait l'objet de très nombreuses publications scientifiques depuis les années 50 (RAPOPORT et CHAMMAH [1965]). Il a été énoncé par Albert W. TUCKER (TUCKER [1950]). Ce jeu fait partie du groupe des jeux sous *forme stratégique* ou *forme normale*.

Le contexte est donné par ces quelques lignes :

Supposons que deux suspects (*Suspect 1* et *Suspect 2*) aient été appréhendés par la police et interrogés dans des pièces séparées dans le cadre d'un vol. La police n'a pas assez de preuves en sa possession pour accuser l'un ou l'autre des suspects du vol. Sans élément complémentaire, les deux suspects écoperont d'un an de prison. Si l'un ou l'autre des suspects dénonce l'autre, le dénoncé écoperera de 10 ans de prison et l'autre sera relaxé. Si les deux suspects se dénoncent mutuellement, ils écoperont de 4 ans de prison chacun.

Si nous modélisons le récit précédent en notant C (comme *Confess* en anglais) lorsque le suspect dénonce et NC (*Non-Confess*) lorsque le suspect ne dénonce pas, nous obtenons la matrice des gains de ce jeu reprise par le tableau 4.1 où chaque case indique pour chaque suspect le gain obtenu en fonction de sa décision.

TABLEAU 4.1 – Matrice des gains : Dilemme du prisonnier

		<i>Suspect 2</i>	
		NC	C
<i>Suspect 1</i>	NC	-1, -1	-10, 0
	C	0, -10	-4, -4

Notre objectif est d'envisager la manière dont les suspects se comporteront dans le jeu donné. Le jeu n'est pas répété. Ainsi, observons les réactions envisageables pour le *Suspect 1*.

- Si le *Suspect 2* ne dénonce pas, alors le *Suspect 1* recevra un gain de 0 en dénonçant et de -1 s'il ne dénonce pas non plus. Donc, il est aussi plus profitable de dénoncer.
- Si le *Suspect 2* a dénoncé, alors le *Suspect 1* recevra un gain de -4 en dénonçant également et -10 s'il ne dénonce pas. Donc, dans ce cas, il est plus profitable de dénoncer.

La stratégie de dénonciation est donc la stratégie la plus appropriée dans les deux cas. On la nomme *stratégie dominante*. Cette stratégie est la plus « favorable » à chaque suspect indépendamment du choix de l'autre suspect.

Le dilemme du prisonnier illustre le conflit entre les incitations sociales à coopérer et les incitations privées à ne pas le faire : chaque prisonnier fait face à un dilemme entre sa rationalité individuelle qui lui dicte de dénoncer son complice et sa rationalité collective qui lui dicte de se taire (AXELROD [2006]).

4.1.3 Les jeux dynamiques

Nous avons préalablement présenté les principales caractéristiques de la théorie des jeux et illustré cette théorie par un jeu sous forme normale. Nous avons en particulier présenté les jeux pour lesquels les joueurs prennent leurs décisions/stratégies de façon simultanée. Il reçoivent à l'issue d'un traitement leur gain conformément à la matrice des gains préalablement présentée. Dans ce cas, les joueurs prennent leurs décisions sans connaissance de la décision des autres joueurs. D'autres jeux, comme les jeux de cartes, les jeux de plateaux ... se jouent dans le temps et tour après tour. Ainsi, chaque joueur prend une décision qui est alors visible par l'ensemble des autres joueurs qui, à leur tour, prennent leurs décisions.

Ces jeux sont appelés *jeux dynamiques* ou *jeux sous forme extensive*. Les décisions des joueurs font également référence à l'historique des décisions jouées. Ils prennent en compte de manière détaillée la structure séquentielle du problème de décision. Cette structure est souvent représentée sous la forme d'un *arbre de jeu* comme présenté, en exemple, sur la figure 4.2.

Pour cet exemple, imaginons qu'il existe deux joueurs, l'entreprise 1 et l'entreprise 2. Chacune d'entre elles tente d'arbitrer entre deux stratégies d'implantation dans deux zones de commercialisation notées A et B. L'entreprise 1 choisit en premier. Si l'entreprise 2 suit l'entreprise 1 dans la même région, alors l'avantage du premier sera de 2/3 du profit potentiel sur la région. L'entreprise 2 recevra pour sa part 1/3 de ce profit. Si l'entreprise 2 se déplace dans l'autre région, alors chaque entreprise obtiendra tout le profit pour leur région respective. La région A a un marché deux fois plus important que la région B. Le total des bénéfices obtenus dans la région A est égal à 12 tandis que dans la région B, ce total est de 6.

La représentation de l'arbre de jeu est donnée par la figure 4.2. L'arbre construit est conçu pour être lu du haut vers le bas. Chaque cercle représente un nœud décisionnel ou une situation finale. Ainsi, le nœud supérieur représente le choix initial de l'entreprise 1, et les deux branches descendantes représentent ses deux options A ou B pour marché A ou marché B. L'entreprise 2 peut également choisir l'option A ou B, encore représentée par des branches descendantes du nœud. Cela conduit à un nœud terminal représentant la fin du jeu. Chaque nœud final reprend le gain défini et connu à l'avance des deux joueurs.

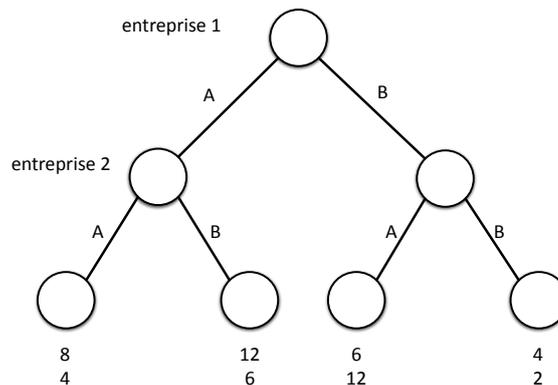


FIGURE 4.2 – Jeu simple sous forme extensive

Chaque joueur connaît le chemin pris par son concurrent au moment de sa décision. Une analyse de la stratégie dominante est réalisable de la même façon sur ce type de jeu.

4.1.4 Les jeux coopératifs

Les jeux coopératifs forment une troisième forme de jeu en théorie des jeux (MOULIN [1991] et GARÈNE [1992]). La théorie des jeux coopératifs (jeux de négociation, jeux coalitionnels) propose des concepts de solution lorsque les individus décident de former des accords : problèmes de répartition des coûts ou des gains, du pouvoir de décision... Ces concepts reposent sur des critères de rationalité individuelle, d'efficacité collective et d'équité. Dans ce type de jeux, les joueurs cherchent à maximiser le gain global au profit du gain individuel. La notion de concurrence ou de victoire individuelle est donc mise au second plan.

Ainsi, pour ces jeux, les joueurs ne seront plus libres de leurs choix. Ils accepteront des stratégies sous format contractuel validées par les autres joueurs.

Nous avons choisi de ne pas approfondir cette approche qui, dans notre contexte de concurrence parfaite, n'est pas applicable.

4.2 Transposition de la théorie des jeux au business game

Le premier chapitre de ce mémoire nous a donné l'occasion de présenter les business games. Les quelques paragraphes précédents nous permettent d'imaginer que les business games peuvent être le théâtre d'expérimentations en théorie des jeux.

Ainsi, nous souhaitons dans ce paragraphe présenter dans un premier temps les rapprochements réalisables et dans un second temps les différences et points de blocages majeurs.

4.2.1 Théorie des jeux et business game

Les business games mettent en scène différentes équipes. Dans la plupart des cas, les concurrents disposent tous, en début de simulation, des mêmes moyens financiers, humains et techniques. Leur mission est de formuler des choix qui, ensemble, forment une stratégie. L'objectif des concurrents est de dominer. Cette domination peut s'exprimer en terme de parts de marché, rentabilité, de rayonnement commercial...

Les premiers éléments énoncés confirment que nous pouvons appliquer dans une première approche la théorie des jeux. Les ingrédients de base sont présents.

Les modalités de prise de décisions doivent être également soulevées. En effet, en début de simulation, les joueurs prennent tous des décisions sans historique ou plutôt avec le même historique. Les décisions sont prises de façon synchronisée en aveugle (sans connaissance de la décision des concurrents). Nous sommes donc en présence d'un jeu sous forme normale.

Dès la simulation du premier tour de jeu, les premiers résultats apparaissent et procurent un historique propre à chaque joueur. Les choix sont répétés. Lors de chaque répétition, les règles évoluent, les choix se diversifient, les positions envisageables changent également. Nous nous trouvons donc dès à présent en face d'un jeu sous forme extensive.

La définition de gain se retrouve également dans les business games. Nous pourrions assimiler le gain aux commandes obtenues sur le jeu ou par couple produit/marché pour être plus précis. En effet, le marché total est défini par le scénario de marché. La somme des commandes est égale au marché. Nous nous retrouvons donc dans une situation de partage ressemblant à la description du jeu sous forme extensive (figure 4.2). Dans ce cas, afin de réaliser une matrice des gains complète, nous devrions calculer pour chaque configuration concurrentielle les commandes et donc l'impact sur le marché.

En première analyse, la théorie des jeux semblerait applicable aux business games qui comportent beaucoup de caractéristiques communes avec les jeux. Cependant, dans la section suivante, nous tenterons de prouver que cette exploitation n'est pas si naturelle que cela et que des verrous seraient à lever.

4.2.2 Les particularités des business games

Stratégies des joueurs non connues de manière exhaustive

En théorie des jeux, il est important de disposer de l'ensemble des stratégies des joueurs. Cette connaissance permet d'évaluer les stratégies dominantes dans la matrice des gains. Cependant, lors des business games, les décisions sont, en partie, prises de manières cachées. Toutes les décisions commerciales sont connues a posteriori par l'ensemble des acteurs économiques et diffusées sous forme d'études de marché. Cependant, les décisions internes comme l'investissement technique, le recrutement, les opérations de recherche et de développement sont masquées. Il devient donc impossible de disposer de l'ensemble des stratégies et donc d'en analyser la domination. Nous nous trouvons dans un cas de matrice des gains de très grande taille à données manquantes.

La résolution de ce type de problème a permis à John Harsanyi d'obtenir le prix Nobel en économie (HARSANYI [2004]). Ces traitements sont néanmoins complexes.

La matrice des gains en perpétuelle évolution

La matrice des gains est aussi un élément difficile à évaluer dans le cadre des business games. En effet, celle-ci est modifiée à chaque tour de jeu et est la résultante de nom-

breuses données connues (historiques, données marketing ...) et inconnues (données techniques, R&D ...).

Aussi, sur le produit UpStrat, l'établissement d'une matrice de gain serait un exercice dont le traitement informatique serait long et fastidieux. Il faudrait simuler l'ensemble des stratégies des concurrents, croiser les performances obtenues, classer les équipes et en extraire les classements. Ces données ne pourraient être traitées en amont de la simulation puisque les concurrents s'adaptent aux contextes antérieurs (chaque fois différents). Ce traitement devrait donc être réalisé en cours d'optimisation et entraînerait un temps de traitement élevé.

La temporalité des décisions

Quelques décisions pour les business games n'ont pas d'impact sur la décision en cours mais offrent des outils de différenciation sur les tours suivants. Ces choix revêtent donc un caractère stratégique majeur nuisant à la performance instantanée mais indispensables aux performances futures. Nous avons d'ailleurs, dans notre proposition de modélisation, intégré cette dimension en projetant les décisions sur les n tours suivants.

Les jeux sous forme extensive intègrent parfaitement l'historique. Le prévisionnel donnerait naissance à un nombre important de nouvelles stratégies. Il serait difficile d'évaluer le gain de ces choix à portée future.

Pour conclure sur ce thème, nous pensons qu'il serait très difficile et pas forcément performant d'aborder l'étude du business game considéré sous l'angle de la théorie des jeux. Nous préférons aborder la problématique de l'anticipation des décisions des équipes par analyse de l'historique et identification de typologies de stratégie. Pour cela, une méthodologie basée sur des méthodes d'extraction de connaissances est proposée.

4.3 Notre approche

Le paragraphe précédent nous permet de douter de la pertinence immédiate du déploiement des modèles de théorie des jeux pour notre cas d'étude. Ainsi, nous souhaitons nous positionner dans un contexte différent, à savoir celui de l'analyse d'historiques par méthodes d'extraction de connaissances. Nous avons été motivés par des travaux d'extraction et de traitement des données qui ont été réalisés par HAO et collab. [2015] et DRACHEN et collab. [2014] sur des jeux de données provenant des *Massively Multiplayer Online Role-Playing Games* (MMORPGs). Sur une banque de données de parties jouées, ils ont souhaité connaître les chances de succès d'un joueur. Les données regroupaient les actions du joueur, les temps de jeu, les orientations des joueurs et les constitutions des équipes. Nous pensons que cette approche est transposable à notre étude.

En effet, nous bénéficions d'un ensemble de simulations jouées dans des environnements similaires. L'analyse de ces données nous permettra d'en extraire de la connaissance. Cette connaissance sera ensuite réintroduite en cours d'optimisation afin de guider l'agent. L'optimisation pourra donc être réalisée sans attendre la décision des concurrents ce qui permettra de respecter le principe de confidentialité et de simultanéité des décisions des équipes et ainsi être réalisée en cours de déroulement d'une partie.

La connaissance pourra être exploitée par les deux rôles confiés à l'agent. Pour son rôle de *concurrent*, son mode d'exploitation a été exposé précédemment. Dans un rôle de *coach*, l'agent pourra proposer des décisions aux équipes qui seront induites par les décisions prises précédemment. Il pourra également anticiper les ventes prévisionnelles d'une équipe au regard de ses décisions. Cette opération est la plus fastidieuse pour les

participants au jeu. Les équipes pourront alors, de façon éclairée, valider ou modifier leurs choix. L'agent pourra également informer l'équipe sur le « chemin décisionnel » qu'elle a pris. Notons que nous ne souhaitons pas anticiper les chances de succès d'une équipe au regard de son cheminement de décisions.

Afin d'analyser les données extraites et d'en déduire des comportements, nous proposons de suivre la méthodologie présentée sur la figure 4.3.

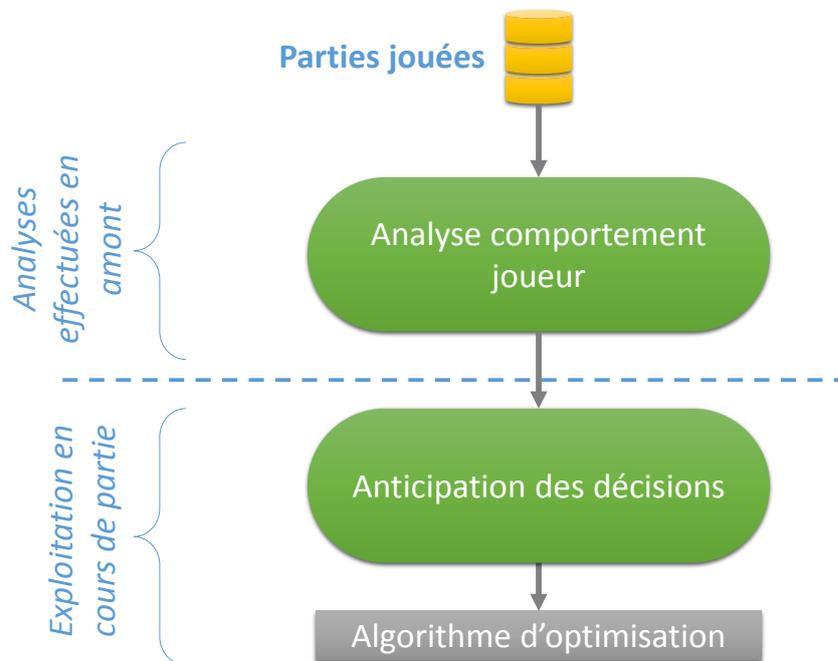


FIGURE 4.3 – Méthodologie générale

Cette méthodologie se décompose en deux phases distinctes :

1. Extraction de connaissances (analyse du comportement des joueurs). Cette étape est faite en amont de l'exploitation. Il s'agit d'une analyse *off-line* faite sur un historique et pouvant être mémorisée à chaque tour de jeu.
2. Exploitation des connaissances extraites en cours de partie (anticipation des décisions des équipes à chaque tour de jeu).

La première phase exploitera des méthodes de *datamining* afin de différencier et caractériser des stratégies de jeu. Dans la phase suivante, nous proposerons une approche originale de parcours des connaissances au moyen d'*arbres d'anticipation*. Cette deuxième étape nous permettra de lancer l'optimisation en estimant les choix des concurrents.

Afin de valider notre approche, nous avons choisi d'exploiter des données provenant d'un autre jeu de simulation, plus simple : **FirStrat**. Pour cette partie, FirStrat a été préféré à UpStrat pour sa simplicité. Cet atout, pour notre étude, nous permettra de mettre en place une méthodologie de traitement et d'analyse des résultats. En effet, le but à moyen terme, sera de déployer cette méthodologie sur d'autres jeux. Nous disposerons ainsi d'une méthodologie « générique ».

Dans un premier temps, nous présenterons FirStrat. Nous introduirons également les données disponibles, de leur extraction à leur prétraitement. Dans un second temps, nous exposerons leur traitement et en particulier la phase de clustering. A l'issue de ce traitement, nous disposerons de groupes d'équipes dont le comportement est similaire. Nous proposerons une méthode originale d'analyse et de suivi des décisions des équipes

au sein d'un groupe sous la forme d'arbre d'anticipation. La construction de ces arbres sera la dernière étape de l'extraction de connaissances. Nous proposerons alors une méthode de classification permettant d'associer une nouvelle équipe (équipe concurrente par exemple) à un groupe déjà établi afin d'en prévoir le comportement futur. Chacune de ces étapes sera détaillée ci-après.

4.4 Le simulateur FirStrat

Comme indiqué en introduction de ce chapitre, nous souhaitons valider nos propositions et l'approche présentée sur un business game moins ambitieux que le simulateur UpStrat. Ainsi, nous nous sommes orientés vers FirStrat que nous décrivons rapidement.

4.4.1 Présentation générale

Le business game FirStrat (STRAT&LOGIC [2000]) est un produit d'initiation à l'entreprise et à la concurrence. Il offre un support performant pour illustrer les fondamentaux de *finance d'entreprise*. Il permet, comme UpStrat, de mettre en concurrence des équipes sur un modèle économique. Cependant, les participants ont beaucoup moins de décisions à prendre et moins d'analyses à réaliser. Ils doivent se positionner sur deux produits au maximum et sur un seul marché. Chaque équipe devra gérer son outil de production, ses stocks de matières premières, de produits finis et sa force de vente.

Les décisions concurrentielles sont :

- le prix,
- la publicité,
- la commission des commerciaux,
- la force de vente,
- le délai de paiement.

Toutes les décisions présentées ont un impact sur le cycle en cours. Pour mémoire, UpStrat offre en plus la possibilité aux équipes d'investir en recherche et développement, cependant les effets de ces actions sont décalés d'une période.

Enfin, un cycle de jeu correspond, dans ce simulateur, à un mois de vie de l'entreprise virtuelle.

De part sa simplicité d'accès et par la thématique pédagogique exposée, FirStrat est un outils très utilisé en formation supérieure : Licences, Masters, écoles de management et écoles d'ingénieurs. Les participants aux sessions ne disposent en général pas de pré-requis.

4.4.2 Présentation des données traitées

Nous disposons de nombreuses bases de données de parties jouées. Afin de ne pas intégrer de biais dans l'analyse des décisions, nous avons sélectionné les parties qui se sont déroulées sur le même scénario de marché. Nous avons isolé les décisions d'un seul produit. Notre base de données exploitée pour les analyses est constituée de 90 parties regroupant 590 équipes sur 12 cycles de jeu au maximum. Chaque cycle regroupe 5 décisions par équipe.

Notons également que les données traitées sur FirStrat sont des données numériques continues. Seul le délai de paiement se traduit par une modalité.

4.4.3 Difficultés à surmonter

L'intégration de connaissances au processus d'optimisation intègre des difficultés que nous résumons dans les paragraphes suivants.

Simultanéité des décisions

Comme présenté précédemment, les MMORPGs sur lesquels des études statistiques ont été menées dans la littérature, bénéficiaient d'un mode de décision séquentiel des participants. Un joueur prend sa décision, le concurrent prend ensuite la sienne en connaissant le mouvement fait par l'opposition et ainsi de suite. Dans notre cas, nous devons faire face à une décision inconnue du concurrent.

Exploitation en cours de partie

Une autre difficulté à surmonter provient du mode de déploiement du système. En effet, celui-ci sera embarqué dans le jeu et devra fonctionner rapidement en limitant la quantité de données à déployer sur différents postes de travail. Nous veillerons donc à maximiser les traitements réalisés en amont. Les traitements dynamiques seront évités ou réduits.

4.5 Préparation des données

La préparation des données est importante. Elle intègre la collecte, le traitement et la sélection des données à traiter. Nous présentons dans les paragraphes suivants les opérations effectuées sur les données à notre disposition.

4.5.1 Collecte et extraction des données

La première étape consiste à cibler les données issues des bases de données. Dans notre cas, nous avons ciblé les données de jeux joués et homogènes. Toutes les parties sélectionnées ont été réalisées sur le même scénario de marché avec les mêmes événements économiques et évidemment sur le même moteur de simulation.

Dans un second temps, nous devons extraire de ces bases de données les variables représentant les décisions des participants. L'objectif de notre démarche n'est pas de prédire le vainqueur d'une partie ou plus généralement d'anticiper la performance finale d'une entreprise au regard de ses décisions passées. Notre souhait est de connaître les décisions futures d'une équipe pour en prédire l'impact sur le marché. C'est pourquoi, nous avons sélectionné uniquement les variables décisionnelles impactant le marché.

Nous disposons donc au départ de fichiers disjoints. La figure 4.4 présente une série de données collectées pour lesquelles chaque jeu a été déroulé respectivement sur k, l et n tours de jeu. Pour chaque jeu, le nombre d'équipes n'est pas non plus garanti. Sur cette même figure nous présentons p, q et m équipes.

Ainsi, l'hétérogénéité s'est manifestée comme cité ci-dessous :

- Le nombre d'équipes inscrites : Les parties comportent de 4 à 10 équipes.
- Le nombre de tours joués : Les parties ont été jouées sur une période variant entre 6 et 10 tours.

Toutes les variables de jeu sont cependant identiques.

Jeu	Equipe	Cycle	Variable 1	Variable 2	...	Variable i	
Jeu 1	1	1					
		...					
		k					
	...	1	1				
			...				
			k				
	p	1	1				
			...				
			k				
Jeu 2	1	1					
		...					
		l					
	...	1	1				
			...				
			l				
	q	1	1				
			...				
			l				
...							
Jeu j	1	1					
		...					
		n					
	...	1	1				
			...				
			n				
	m	1	1				
			...				
			n				

FIGURE 4.4 – Données brutes extraites

4.5.2 Linéarisation des données

Chaque jeu joué se compose d'un ensemble de données classées par cycle, équipe, produit et zone de jeu. Ces données exportées doivent être linéarisées afin de disposer d'une ligne par équipe regroupant toutes les décisions des cycles joués. Chaque ligne représentera donc toutes les décisions concurrentielles prises au cours du jeu par une équipe pour le produit considéré.

Nous obtenons ainsi une matrice présentant autant de lignes que d'équipes inscrites. Le nombre de colonnes de la matrice correspond au nombre de variables décisionnelles étudiées sur un tour multiplié par le nombre de tours de jeu joués.

Pour rappel, les variables sont :

- *Prix* pour le prix du produit,
- *Publicité* pour le budget publicitaire alloué spécifiquement au produit,
- *Commission* pour la commission des commerciaux sur le produit,
- *NbFV* pour l'effectif de la force de vente,
- *DP* pour le délai de paiement imposé au client.

Une colonne pourra ainsi porter le nom suivant : *Prix_Cycle_k* pour le prix donné sur le cycle *k*.

La figure 4.5 présente cette transformation.

L'information contenue dans la variable jeu n'est pas conservée car l'appartenance d'une équipe à un jeu n'offre pas d'information complémentaire sur le comportement de cette équipe.

Nous ajoutons toutes les équipes de toutes les parties sélectionnées à la matrice générale.

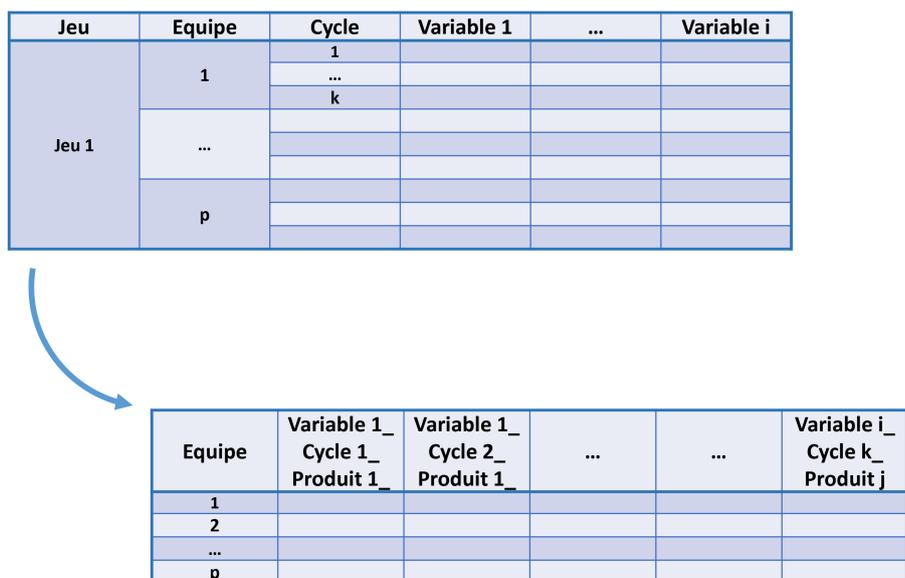


FIGURE 4.5 – Linéarisation des décisions

4.5.3 Sélection des données

Nous sélectionnons les variables comprises entre les cycles n et p soit $[n, \dots, p]$. Nous nommerons cet intervalle : « fenêtre ».

Le premier cycle de jeu ne sera jamais intégré à la sélection de données afin d'éviter les erreurs de compréhension des règles du jeu des participants.

Dans le même esprit, nous ne retenons pas non plus le dernier tour de jeu. Celui-ci est régulièrement le théâtre d'optimisations communément appelées *stratégie de fin de jeu*. Ces comportements, bien souvent non cohérents sur le long terme, visent à améliorer rapidement les performances des entreprises pilotées au détriment d'un développement futur et pérenne.

Enfin, nous maintiendrons une largeur de sélection de données suffisante pour une prédiction efficace. Ainsi, chaque fenêtre contiendra au moins trois cycles de décisions comme indiqué dans la figure 4.6. Cette sélection assure un suivi et une cohérence du comportement des équipes.

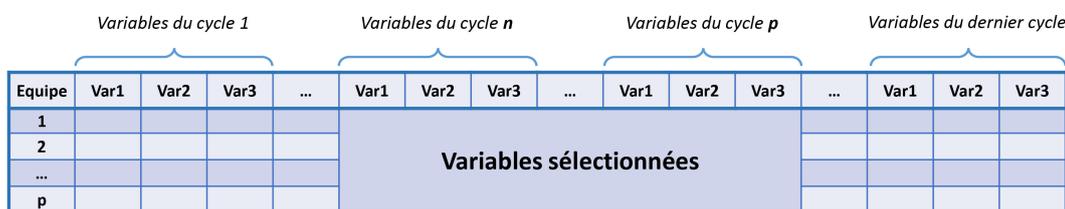


FIGURE 4.6 – Fenêtre de sélection de variables

4.5.4 Pré-traitement des données

Nous souhaitons éliminer toutes les variables non significatives ou statistiquement hors norme de la matrice précédemment générée.

Nous procédons dans un premier temps à la suppression des colonnes dont l'écart-type est nul pour éliminer les variables sans intérêt, ne permettant pas de différencier les

comportements.

Nous avons souhaité extraire tous les individus qui présentaient des caractères anormaux. Ces comportements proviennent d'incompréhensions des règles du jeu ou d'une tentative de déstabilisation de l'univers économique. Ils se caractérisent par des décisions marginales.

Nous exploitons une technique de détection des *outliers* sur chaque variable et calculons le *ScoreZ* de chacune d'elles. Nous considérerons que si un individu présente une variable statistiquement « anormale » alors il sera écarté de la matrice à traiter et donc du traitement.

Identification des outliers par le Score Z

Ainsi pour chaque valeur y_i de la variable y , le Score Z de la variable (z_i) s'obtient par la formule (avec σ_y l'écart-type de y) :

$$z_i = \frac{y_i - \bar{y}}{\sigma_y} \quad (4.1)$$

Iglewicz et Hoaglin proposent d'exploiter le score Z évolué (IGLEWICZ et HOAGLIN [1993]). Il se traduit par la formule :

$$m_i = \frac{0.6745(y_i - \bar{y})}{MAD_y} \quad (4.2)$$

où MAD_y est la déviation absolue de la médiane. Elle se calcule par la formule suivante :

$$MAD_y = \text{Mediane}(|y_i - \bar{y}|) \quad (4.3)$$

Iglewicz and Hoaglin ont montré qu'une valeur m_i supérieure à 3,5 obtenue pour une seule variable d'un individu i indique que l'individu est très probablement un outlier. Ainsi, la première variable en dehors de l'intervalle de tolérance entraîne l'exclusion complète de l'équipe.

4.5.5 Centrage des données

Afin d'annihiler l'effet de grandeur des variables à traiter, nous centrons l'ensemble des variables de la matrice en appliquant :

$$x'_i = \frac{x_i - \min(x)}{\max(x) - \min(x)} \quad (4.4)$$

Notons que nous devons réaliser, dans la phase d'exploitation, l'opération inverse. Ainsi, nous stockerons pour chaque variable les informations nécessaires à l'opération.

4.6 Datamining : Partitionnement des équipes par clustering

La phase d'extraction de connaissances a pour objectif d'analyser le comportement des équipes au moyen de l'historique de leurs décisions. Cette opération est réalisée en amont du processus d'optimisation. La figure 4.7 présente les trois actions menées pendant cette analyse. Les données sont d'abord extraites. Ensuite, elles sont transformées.

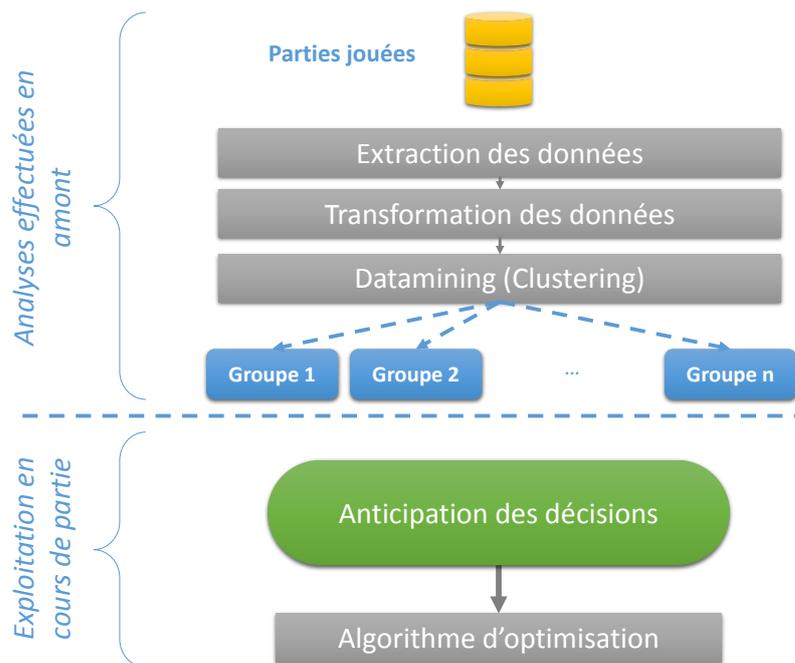


FIGURE 4.7 – Détail de la première phase de l'approche

L'objectif de la dernière étape de ces analyses effectuées en amont est donc d'identifier des comportements similaires de joueurs pendant la durée d'une simulation. Ainsi, nous proposons d'aborder cette problématique à l'aide du clustering.

Le clustering est une technique d'analyse de données très développée. Elle permet d'extraire de la connaissance des données préparées. Cette technique permet de grouper des individus présentant des attributs semblables ou comparables. Il existe de nombreux types d'attributs comparables. Cependant, dans notre étude sur ce chapitre et donc sur FirStrat, nous ne traiterons que des attributs *quantitatifs*. Ils prennent une forme numérique. On retrouve par exemple le délai de paiement, l'effectif en force de vente... Les statistiques univariées peuvent être calculées sans transformation.

Il existe de nombreuses méthodes de clustering dans la littérature. HAN et collab. [2011] présente une décomposition des méthodes en diverses catégories :

- Méthodes de partitionnement (K-medoids, K-means...)
- Méthodes hiérarchiques (Agglomerative Algorithms, Divisive Algorithms)
- Algorithmes basés sur la densité (Density-Based Connectivity Clustering, Density Functions Clustering)
- Méthodes à base de grilles
- Méthodes basées sur la co-occurrence de données catégorielles
- Clustering basé sur les contraintes
- Algorithmes de clustering utilisés en Machine Learning et optimisation,
- Algorithmes de clustering évolutif
- Algorithmes de clustering pour données de grande dimension

Les précédentes études (HAO et collab. [2015] et DRACHEN et collab. [2014]) nous ont montré que les méthodes de partitionnement et les méthodes hiérarchiques ont donné de très bons résultats sur les MMORPGs. Ainsi, nous avons souhaité exploiter ces mêmes méthodes pour nos jeux de données.

4.6.1 Description de l'opération de clustering

Le but du clustering est de rassembler des individus en groupes d'individus similaires. La figure 4.8 représente, sur un graphique en deux dimensions, une illustration d'un clustering permettant de mettre en évidence trois clusters (MATLAB [2010]). Cette illustration a été générée à titre d'exemple à partir de données préparées. Les croix situées au sein des groupements de points représentent le *centroid* de chaque cluster.

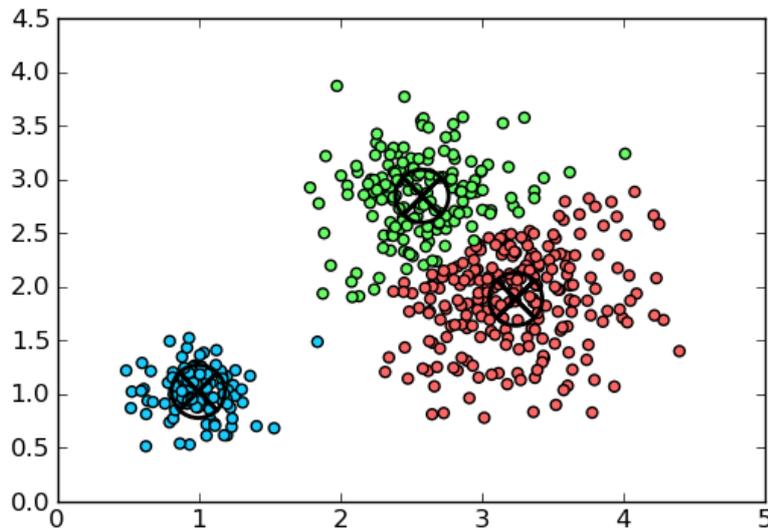


FIGURE 4.8 – Exemple de clustering

On définit ainsi l'opération de clustering pour un ensemble D de n échantillons $D = \{x(1), \dots, x(n)\}$ dans un espace de dimension d . D est partitionné en un ensemble de k sous-groupes disjoints D_j tels que :

$$D = \cup_{j=1}^k D_j \quad (4.5)$$

où :

- $D_i \cap D_j = \emptyset, \quad \forall i \neq j,$
- Tous les éléments d'un sous-groupe sont similaires par rapport à un critère Φ .

Une partition π est notée $\pi = (D_1, D_2, \dots, D_k)$. Le problème de clustering s'écrit alors :

$$\pi^* = \operatorname{argmin}_{\pi} f(\pi) \quad (4.6)$$

où f est une fonction d'optimisation ou de pilotage du critère Φ .

Dans la suite de ce chapitre, nous présentons les méthodes de clustering classiques :

- Méthodes par partitionnement.
- Méthodes hiérarchiques.

A partir de ces méthodes, de nombreuses variantes sont nées.

Les méthodes de partitionnement

Les méthodes de partitionnement ont été mises en place par FORGY [1965]. Leur principe est de diviser un ensemble d'individus en K ensembles de points disjoints. L'homogénéité des groupes générés est assurée par la définition d'une fonction de *dissimilarité*

qu'il faut minimiser. Souvent cette fonction est basée sur la distance notée d entre les individus d'un groupe.

Dans toutes les étapes de clustering, la notion de distance est fondamentale. Il existe de nombreuses approches proposant des modèles de calcul de distances très différents. Dans une première approche, nous avons choisi d'exploiter la distance Euclidienne.

Lors d'un clustering, chaque groupe d'individus formé sera représenté par un *centroïd* qui pourra être un des éléments de l'ensemble ou un élément calculé.

L'algorithme le plus fréquemment utilisé pour cette méthode est K-means de MAC-QUEEN et collab. [1967] présenté par l'algorithme 4.1.

Algorithm 4.1 Algorithme K-means classique

```
for  $k = 1$  to  $K$  do
  Prendre  $r_k$  aléatoirement parmi les points de  $D$ 
end for
while Le cluster  $C_k$  est modifié do
  for  $k = 1, \dots, K$  do
     $C_k = \{x \in D, d(r_k, x) \leq d(r_j, x)\} \forall j = 1, \dots, K \text{ et } j \neq k$ 
  end for
  for  $k = 1, \dots, K$  do
    Construire un nouveau  $r_k$  qui est la moyenne des points de  $C_k$ 
  end for
end while
```

Cet algorithme est itératif. Il démarre par une sélection de k individus de D qui représenteront les premiers centroïds des clusters C_k . Les individus sont ajoutés au cluster dont le centroïd est le plus proche. Dans l'étape suivante, on remplace les k centroïds arbitrairement définis par les barycentres des clusters constitués. On compare ensuite la stabilité des centres de chaque cluster. Si celle-ci est atteinte, on stoppe l'algorithme dans le cas contraire, on réaffecte les individus au cluster de centroïd le plus proche.

Les méthodes hiérarchiques

Les méthodes de clustering hiérarchique sont basées sur le regroupement d'individus en fonction d'une indication de distance. Il s'agit alors de méthodes agglomératives. L'algorithme 4.2, ci-dessous, en présente le principe.

Une méthode par division existe également. Son principe de fonctionnement est totalement inverse. Elle démarre d'un seul groupe puis par division construit des clusters.

Algorithm 4.2 Méthode hiérarchique classique

```
for  $i = 1, \dots, n$  do
  while qu'il reste plusieurs clusters do
    for  $k = 1, \dots, K$  do
      Prendre  $C_i$  et  $C_j$  afin de minimiser la distance  $D(C_k, C_h)$  entre deux clusters
       $C_i \leftarrow C_i \cup C_j$ 
      Supprimer le cluster  $C_j$ 
    end for
  end while
end for
```

Le principe de la méthode est donc le suivant :

1. les clusters initiaux sont les individus,
2. on calcule les distances entre les clusters,
3. les deux clusters les plus proches sont fusionnés et remplacés par un seul,
4. on reprend en 2. jusqu'à n'avoir plus qu'un seul cluster qui contient tous les individus.

Le clustering hiérarchique donne naissance à un *dendrogramme* comme présenté dans la figure 4.9. Ce dendrogramme a été réalisé avec des données générées aléatoirement en guise d'exemple (MATLAB [2010]).

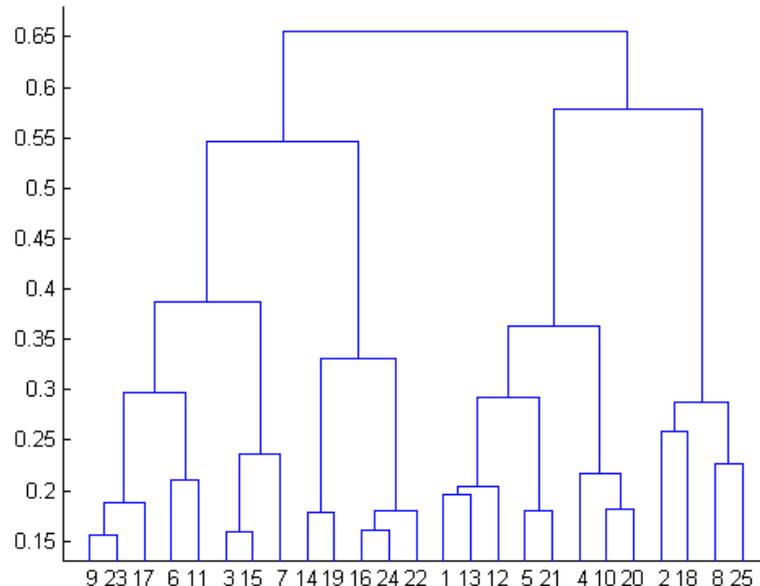


FIGURE 4.9 – Exemple de dendrogramme

La méthode la plus classique pour ce type de clustering est la méthode de *Ward* (WARD JR [1963]).

La formule de distance dans l'algorithme de Ward pour deux clusters A et B, de centroïdes Cn_a et Cn_b , et d'effectifs n_A et n_B est :

$$d(A, B) = \frac{d(Cn_a, Cn_b)^2}{\frac{1}{n_A} + \frac{1}{n_B}} \quad (4.7)$$

4.6.2 Qualité d'un clustering

Le clustering est un problème d'extraction de connaissances non supervisé. Il n'est donc pas possible de connaître la qualité absolue d'un partitionnement. Cependant deux critères importants permettent d'en évaluer la pertinence :

- un critère de **compacité** : les individus d'un cluster doivent être les plus semblables possibles.
- un critère de **séparation** : les clusters doivent être les plus différents possibles. La distance entre ces clusters doit donc être la plus grande possible.

Il existe de nombreux indicateurs de performance que l'on peut retrouver par exemple dans l'article BEN-DAVID et ACKERMAN [2009]. Cependant, nous présentons uniquement les critères de performance que nous avons exploités dans notre étude.

Critères de compacité

Variance intra-cluster La variance intra-cluster évalue la similarité entre les éléments d'un cluster C. Elle doit être minimale.

$$Var_{Intra}(C) = \sum_{C_k \in C} \sum_{i \in C_k} d(i, Cn_k)^2 \quad (4.8)$$

où

- Cn_k est le centroïd du cluster k .
- C_k est le cluster k .
- $d(i, j)$ est la distance entre les individus i et j .

Connectivité La connectivité doit être la plus faible possible car elle s'assure que deux individus proches sont classés dans le même cluster.

$$Connectivity = \sum_{i=1}^N \sum_{j=1, j \neq i}^N x_{i, nn_{j,i}} \quad (4.9)$$

où :

- $nn_{j,i}$ correspond au $j^{\text{ème}}$ plus proche voisin de l'individu i
- $x_{i, nn_{j,i}}$ est égal à 0 si i et $nn_{j,i}$ sont dans le même cluster et à $\frac{1}{j}$ dans les autres cas
- N Nombre d'individus

Critère de séparation : variance inter-cluster

La variance inter-cluster évalue l'éloignement entre deux clusters. Elle doit être maximale.

$$Var_{Inter}(C) = \sum_{C_k \in C} \sum_{C_h \in C, h \neq k} d(Cn_h, Cn_k)^2 \quad (4.10)$$

avec Cn_k centroïd du cluster k .

Indicateurs agrégés

Ces indicateurs essaient de combiner les deux dimensions fondamentales caractérisant la qualité d'un clustering en un seul indicateur.

Index de Calinski-Harabasz L'indice Calinski-Harabasz (CH) est calculé par l'équation (4.11) (CALIŃSKI et HARABASZ [1974]).

$$CH(K) = \frac{B}{W} * \frac{N - k}{k - 1} \quad (4.11)$$

où k est le nombre de clusters, N le nombre d'observations.

avec B comme indicateur de séparation

$$B = \sum_{i=1}^k T_i \|Cn_i - m\|^2 \quad (4.12)$$

où T_i est l'effectif du cluster i , Cn_i le centroïd du cluster i et m la moyenne générale des données.

et W l'indicateur de compacité

$$W = \sum_{i=1}^k \sum_{x \in C_i} \|x - Cn_i\|^2 \quad (4.13)$$

où k est le nombre de clusters, x une observation, C_i le $i^{\text{ème}}$ cluster et Cn_i le centroïd du cluster i . Toutes les distances calculées sont les distances Euclidiennes.

L'index de Calinski-Harabasz doit être **maximal**.

Index de Davies-Bouldin Dans l'index Davies-Bouldin (DAVIES et BOULDIN [1979]) est obtenu par les équations suivantes.

L'équation 4.14 mesure la dispersion au sein d'un cluster C_i .

$$S_i = \left(\frac{1}{T_i} \sum_{j=1}^{T_i} |x_j - Cn_i|^p \right)^{\frac{1}{p}} \quad (4.14)$$

où :

- Cn_i centroïd du cluster i
- T_i cardinal du cluster i

L'équation de séparation entre deux clusters C_i et C_j est donnée par l'équation (4.15).

$$M_{i,j} = \left(\sum_{k=1}^n |x_{k,i} - x_{k,j}|^p \right)^{\frac{1}{p}} \quad (4.15)$$

où :

- $x_{k,i}$ est le $k^{\text{ème}}$ élément du cluster i
- n est le cardinal minimal des clusters i et j

Ainsi, pour deux clusters i et j , optimiser la qualité du clustering revient à maximiser $M_{i,j}$ et minimiser S_i . C'est pourquoi, l'index de Davies-Bouldin est défini comme le ratio de S_i par $M_{i,j}$. Ce ratio est donné par l'équation (4.16).

$$R_{i,j} = \frac{S_i + S_j}{M_{i,j}} \quad (4.16)$$

$R_{i,j}$ respecte les propriétés suivantes :

1. $R_{i,j} \geq 0$
2. $R_{i,j} = R_{j,i}$
3. Si $S_j \geq S_k$ et $M_{i,j} = M_{i,k}$ alors $R_{i,j} > R_{i,k}$
4. Si $S_j = S_k$ et $M_{i,j} \leq M_{i,k}$ alors $R_{i,j} > R_{i,k}$

Dans un contexte à N clusters posons $D_i = \max(R_{i,j}), \forall i \neq j, j \in N$. L'index de Davies-Bouldin (DB) se calcule alors par l'équation 4.17.

$$DB = \frac{1}{N} \sum_{i=1}^N D_i \quad (4.17)$$

L'index DB doit être **minimal**.

Intervention d'un expert

En addition des indicateurs présentés dans les paragraphes précédents, le choix du clustering peut être influencé par l'intervention d'un expert. Cette intervention permettra de définir le nombre de classes à conserver. Ce choix sera opéré par rapport à l'expérience de l'expert. Cette option devient très utile lorsque les indicateurs chiffrés donnent des valeurs assez proches les unes des autres.

L'intervention d'un expert permet en outre de ne pas conserver de partitionnement qui présenterait des clusters comportant peu d'individus ou ne présentant pas un caractère marqué (distance trop faible par rapport au cluster voisin).

Dans certains cas, le nombre de clusters est défini afin qu'il soit possible de nommer clairement le « caractère » des individus qui les composent. C'est ce qui a été fait pour l'analyse des *MMORPGs*. Nous n'avons pas souhaité identifier aussi clairement les clusters dans notre étude. Nous ne voulions pas influencer le comportement des joueurs lors de la communication de ce type d'information.

4.6.3 Mise en œuvre sur FirStrat

Après avoir présenté quelques éléments généraux sur les méthodes de clustering, nous allons l'expérimenter sur les données FirStrat afin de définir la méthodologie à appliquer. Nous pourrions ainsi faire des choix sur la méthode de clustering à utiliser pour ce type de problématique. Nous définirons également le nombre optimal de clusters au regard d'indices de pertinence.

Présentation des données et des expérimentations réalisées sur FirStrat

Nous présentons les données que nous avons obtenues pour les expérimentations réalisées. Nous disposons d'une matrice composée des données de 590 équipes. Cette matrice regroupe également 60 variables décisionnelles au plus.

Les fenêtres de sélection de données que nous avons réalisées sont les suivantes :

- A partir du cycle 2 : 2-4, 2-5, 2-6, 2-7, 2-8, 2-9
- A partir du cycle 3 : 3-5, 3-6, 3-7, 3-8, 3-9
- A partir du cycle 4 : 4-6, 4-7, 4-8, 4-9
- A partir du cycle 5 : 5-7, 5-8, 5-9

Le nombre de variables s'obtient en multipliant la taille de la fenêtre par le nombre de variables par tour de jeu. Dans notre étude sur la simulation FirStrat, il y a **cinq** variables par tour de jeu.

Aussi, pour chaque fenêtre de données, nous avons déployé deux algorithmes de clustering :

- Clustering hiérarchique : Méthode *D.Ward*
- Clustering par partitionnement : Méthode *k.means*

Les démarches effectuées dans la suite nous ont, tout d'abord, permis de fixer la fenêtre de décisions qui maximisera la pertinence du partitionnement obtenu. Les paramètres qui impactent cette pertinence sont le nombre de clusters, la taille de la fenêtre de décisions et le début de cette fenêtre (tour 2, 3...). Ainsi, le clustering a été réalisé pour des configurations allant de 2 à 7 clusters. Nous avons aussi essayé de donner une interprétation aux clusters identifiés en comparant le contenu de chaque cluster et donc en représentant l'évolution des variables pour chaque tour. Cette interprétation nous permettra

de comparer les résultats issus des opérations mathématiques aux ressentis d'animateurs experts.

Recherche de la meilleure fenêtre de décision

Le tableau 4.2 reporte sur chaque fenêtre de décision, pour les deux indices de qualité retenus et pour les deux méthodes de clustering, le meilleur partitionnement obtenu. A droite de la valeur de l'indice de qualité et entre parenthèses figure le nombre de clusters correspondant à l'indice. Nous avons marqué en gras les configurations que nous souhaitons conserver dans la suite de nos travaux.

TABLEAU 4.2 – Comparaison des clustering par fenêtre de décisions

	CH (max)		DB (min)	
	D.Ward	K.means	D.Ward	K.means
2-4	160,6 (3)	198,1 (3)	1,89 (4)	1,70 (3)
2-5	163,6 (3)	200,1 (3)	1,96 (4)	1,71 (3)
2-6	169,9 (3)	187,3 (3)	1,60 (3)	1,70 (3)
2-7	147,7 (3)	176,6 (3)	1,55 (3)	1,70 (3)
2-8	154,2 (3)	191,5 (3)	1,66 (3)	1,63 (3)
2-9	102,6 (3)	116,0 (3)	1,77 (3)	1,74 (3)
3-5	162,4 (3)	217,2 (3)	1,64 (8)	1,67 (6)
3-6	167,3 (3)	203,2 (3)	1,81 (3)	1,64 (3)
3-7	150,2 (3)	188,2 (3)	1,59 (4)	1,67 (3)
3-8	188,1 (3)	205,9 (3)	1,57 (3)	1,56 (3)
3-9	103,1 (3)	124,9 (3)	1,59 (3)	1,71 (3)
4-6	189,0 (3)	215,9 (3)	1,56 (4)	1,50 (8)
4-7	170,9 (3)	198,1 (3)	1,57 (3)	1,61 (3)
4-8	186,1 (3)	220,6 (3)	1,49 (3)	1,54 (3)
4-9	111,8 (3)	125,0 (3)	1,98 (5)	1,73 (6)
5-7	183,4 (3)	216,0 (3)	1,47 (8)	1,55 (3)
5-8	221,7 (3)	239,3 (3)	1,40 (3)	1,45 (3)
5-9	123,3 (3)	144,0 (3)	1,58 (3)	1,55 (3)

Choix de la méthode de clustering

Les résultats obtenus précédemment nous permettent de faire un choix concernant la méthode de clustering. Les deux critères de performance mettent en avant les mêmes configurations. Le choix de la fenêtre de décisions peut encore être discuté. Cependant, afin d'éviter les bruits obtenus par des décisions de fin de jeu ou par une découverte hésitante des règles en début de jeu, nous proposons de nous orienter vers la fenêtre 5 – 8. Aussi, nous conserverons le partitionnement en **trois clusters**. Enfin, nous nous orienterons vers la méthode de clustering nommée **méthode de partitionnement**. Cette méthode est souvent exploitée dans les analyses analogues (HAO et collab. [2015] et DRACHEN et collab. [2014]).

Interprétation du partitionnement obtenu

Dans cette partie, nous avons souhaité analyser les équipes regroupées dans chacun des clusters. Cette opération a pour but de vérifier si, pour un expert, il est possible de

formuler quelques conclusions sur le comportement des équipes pendant une simulation permettant de valider le partitionnement réalisé de façon non supervisée.

L'objectif n'est pas de définir si une équipe a une chance de gagner ou au contraire, se met en difficulté par rapport au résultat final car cela est très dépendant de son contexte de jeu et en particulier de ses concurrents. Nous cherchons cependant à caractériser les clusters et évaluer si possible les comportements que l'on retrouve dans une partie classique.

Notons que cette opération peut se réaliser sur FirStrat, car sur cette étude, nous nous sommes focalisés sur un seul segment marketing afin de valider la démarche.

Description des clusters Dans un premier temps, nous présentons dans le tableau 4.3 les effectifs par cluster.

TABLEAU 4.3 – Effectif par cluster - Partitionnement en 3 clusters sur la fenêtre 5-8

	Effectif	%
Cluster 1	82	13,9 %
Cluster 2	308	52,2 %
Cluster 3	200	33,9 %
Total	590	100,0 %

Nous remarquons que le cluster 2 est majoritaire dans la représentation des équipes. Le cluster 3 est présent dans un tiers des cas. Le cluster 1 se retrouve pour sa part assez peu représenté (1 cas sur 7 environ).

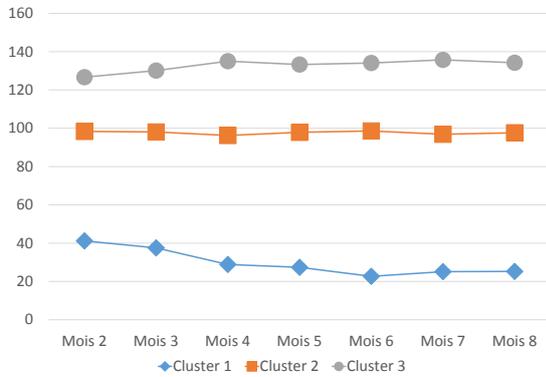
Les figures 4.10a, 4.10b, 4.10c, 4.10d et 4.10e représentent les évolutions des centroïdes des trois clusters identifiés pendant les décisions 2 à 8 selon chacune des 5 variables. Les données présentées sont des indices base 100. La base 100 représente la moyenne générale des données exploitées pour le partitionnement et pour chaque variable à chaque tour de jeu (mois).

Les comportements de chaque cluster sont très caractérisés. En effet, le *cluster 3* présente des valeurs qui sont toujours supérieures aux valeurs moyennes (valeur 100). Ainsi, ces équipes ont tendance à vendre leur produit un peu plus cher que les autres. Pour compenser cette perte de compétitivité commerciale, elles réalisent d'importantes dépenses en publicité et augmentent leurs ressources pour la force de vente (aussi bien en nombre de commerciaux qu'en commission). Le cluster 3 souhaite se démarquer de ses concurrents et présente une stratégie proche de la **différenciation**.

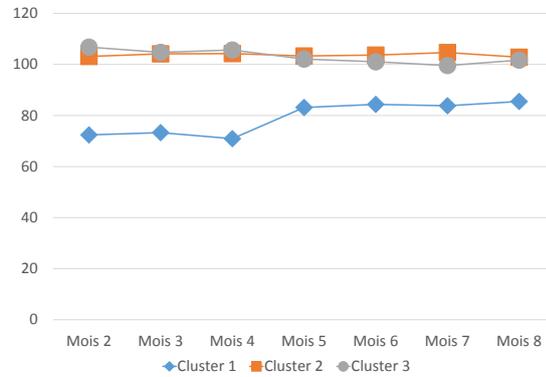
Le cluster 1, pour sa part, présente totalement le comportement inverse. Les prix sont très compétitifs et laissent très peu de marge pour effectuer d'autres actions commerciales. Ainsi, les budgets de publicité sont faibles. Les effectifs en force de vente sont réduits et leur commission est rabotée. Le cluster 1 est très agressif. Il vise certainement une stratégie plus orientée vers le **volume/coût**.

Le cluster 2 est constitué d'un ensemble d'équipes qui prennent peu de risques (au regard des concurrents). Ces équipes restent dans des moyennes données en début de jeu. Nous pourrions décrire ce cluster comme étant un ensemble de **suiveurs**.

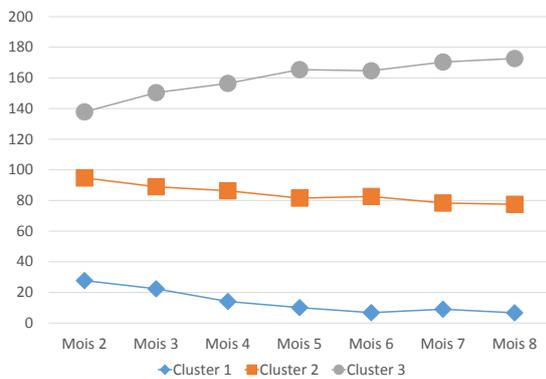
La variable « Délai de paiement » ne permet pas de différencier énormément les équipes. Les clusters 2 et 3 prennent des décisions très similaires. Le cluster 1, quant à lui, impose à ses clients un délai de paiement court. Cette orientation peut sembler logique car il est fort probable que les équipes du cluster 1 captent beaucoup de commandes et subissent les conséquences des fortes croissances soit un niveau de trésorerie faible. La maîtrise de besoin en fonds de roulement est donc une bonne démarche.



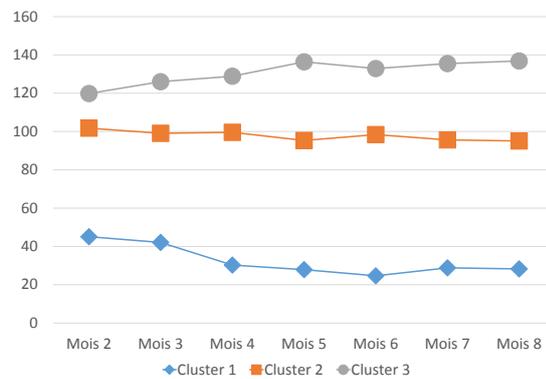
(a) Évolution de la variable **Prix**



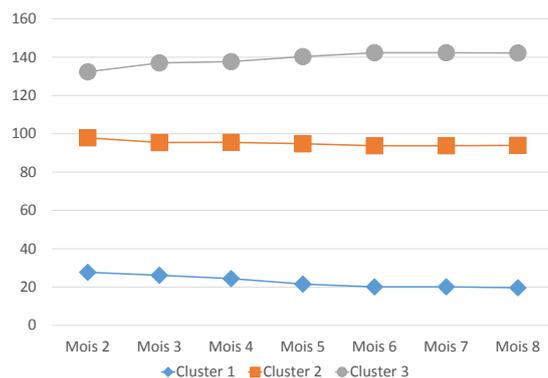
(b) Évolution de la variable **Délai de paiement**



(c) Évolution de la variable **Publicité**



(d) Évolution de la variable **Commission**



(e) Évolution de la variable **Nombre de commerciaux**

FIGURE 4.10 – Valeur moyenne des variables décisionnelles par cluster sur FirStrat

La matrice du modèle de Porter (PORTER et DE LAVERGNE [2003]) présenté ci-dessous (Figure 4.11) nous permet de positionner les clusters 1 et 3. Le cluster 2 figure sur ce modèle à l'intersection des zones car les équipes n'ont pas fait de vrai choix. Notons que sur la simulation étudiée, les stratégies de concentration (fondée sur les coûts ou la différenciation) ne sont pas possibles car le simulateur n'offre pas cette dimension dans les décisions (la dimension Recherche et Développement est absente). Il n'y a pas de segment de marché différent.

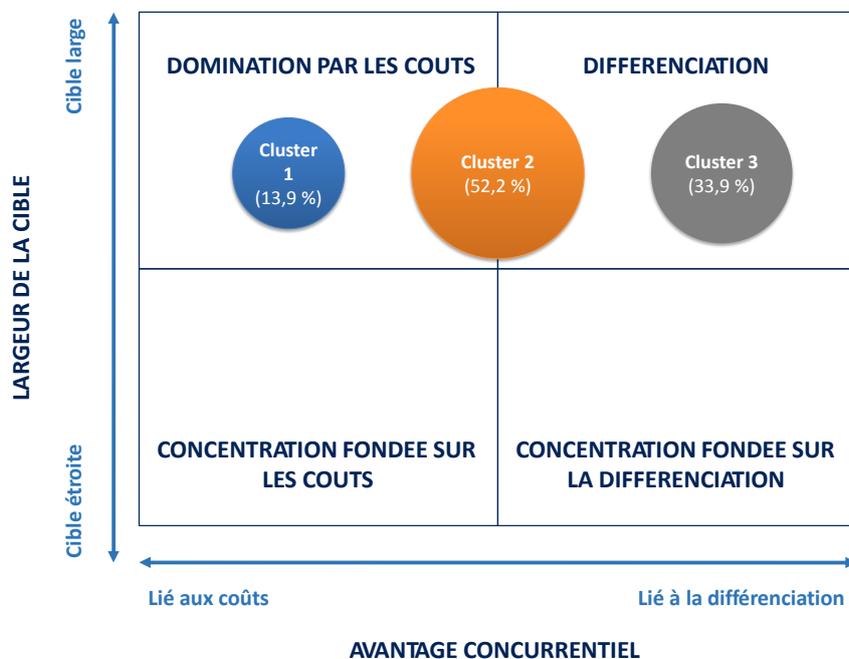


FIGURE 4.11 – Positionnement des comportements de joueurs dans la matrice de M. Porter

Comparaison des observations à l'avis d'experts Afin de confronter nos observations à la réalité du terrain, nous avons souhaité présenter ces conclusions à quelques formateurs avertis sur le simulateur FirStrat.

Le retour formulé par ces formateurs est qu'ils identifient régulièrement des comportements d'équipes similaires à ceux décrits dans le paragraphe précédent. En effet, en général, sur chaque univers une ou plusieurs équipes se positionnent clairement dans un *business model low cost* ou volume/coût au sens de Porter. Elles absorbent de nombreuses commandes en baissant les prix de vente et en misant sur l'amortissement des charges fixes par la majoration des volumes de ventes découlant de ce choix.

Les réactions des concurrents peuvent alors être :

- une baisse de prix afin de suivre les précurseurs dans le modèle low cost et ainsi espérer également une plus grande part de marché,
- une augmentation des délais de paiement, de la rémunération des commerciaux pour adopter un positionnement commercial alternatif dans lequel le service l'emporte sur le prix pour une partie des consommateurs.

Il est difficile dans un univers de faire coexister plusieurs équipes avec une stratégie de low-cost (HILL [1988]). En effet, une prolifération de ce comportement entraînera nécessairement une baisse sensible des prix de marché. Aussi, l'élasticité au prix de la demande des consommateurs ne permettra pas d'accroître suffisamment la demande globale du

marché. Ainsi, les volumes récoltés par les entreprises seraient insuffisants à la couverture des frais fixes et donc sujets à la mise en difficulté des structures (d'où seulement 13,9% d'équipes dans le cluster 1). Par conséquent, les équipes suiveuses représentent une majorité de cas (52,2%) et les sociétés qui prônent la différenciation ne dépassent pas plus d'une équipe sur trois.

4.7 Classification des équipes

Nous disposons à ce stade de clusters d'équipes, calculés selon des données d'historique de parties. Nous les considérons maintenant comme des classes de joueurs. Notre objectif, comme rappelé par la figure 4.12 est de positionner une nouvelle équipe dans la bonne classe en cours de partie. Ces traitements se feront en mode embarqué.

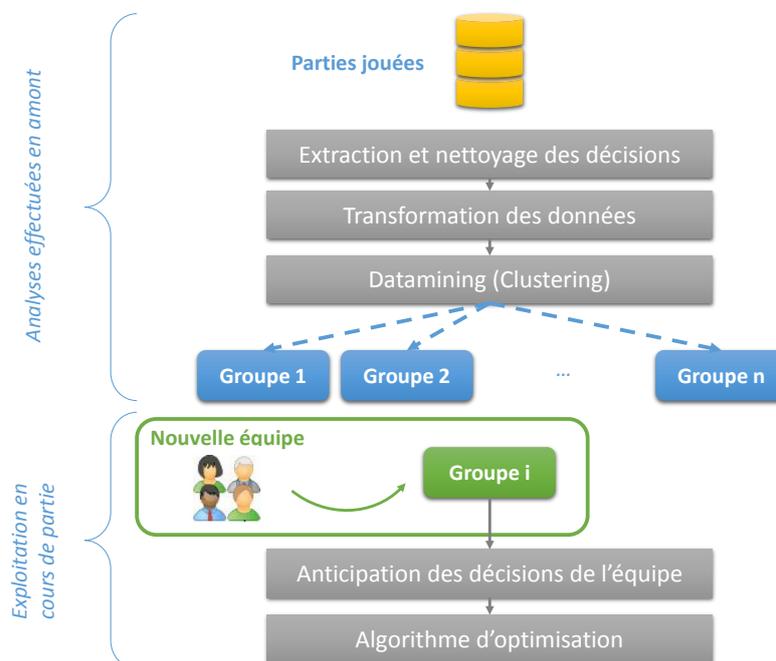


FIGURE 4.12 – Méthodologie générale

Cette partie se consacre donc au design d'une méthode de classification adaptée à notre problématique. Cette étape intervient après le clustering et permet d'affecter une classe à une nouvelle équipe. Comme le montre la figure 4.13 (DHAENENS et JOURDAN [2016]), certaines méthodes de classification imposent la construction en amont d'un classifieur (Phase 1 sur la figure). Ce classifieur est ensuite exploité dans la phase 2.

Après avoir présenté quelques généralités sur la classification, nous présenterons les différentes méthodes de classification expérimentées ainsi que les résultats des expérimentations menées sur les données classées.

4.7.1 Généralités sur la classification

La classification permet de prédire si une instance de données est membre d'un groupe ou d'une classe prédéfinie. Chaque instance peut être représentée par p variables prédictives et une variable cible à prédire. La classification est un apprentissage supervisé car les classes sont connues à l'avance.

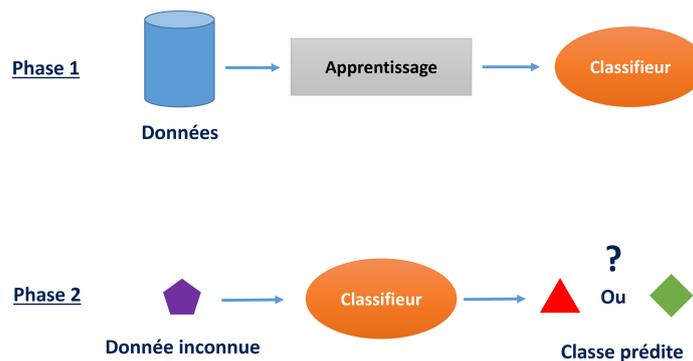


FIGURE 4.13 – Classification

La classification est réalisée par un *classifieur*. La nature de ces *classifieurs* peut être différente : il peut par exemple s’agir d’arbres de décisions, de suites de règles ou encore provenir d’une comparaison de proches en proches.

Nous proposons de présenter différentes méthodes :

- des méthodes construisant un classifieur (modèle de classification) : Régression logistique multinomiale, arbres de décision.
- des méthodes déterminant l’appartenance à une classe sans construction de classifieur : Proximité au centroïde de la classe, Méthode des k plus proches voisins.

Nous présenterons également les arbres de décisions afin de faire le parallèle avec les arbres d’anticipation que nous proposerons en dernière partie de ce chapitre.

4.7.2 Phase 1 - Construction d’un classifieur

L’objectif de cette phase d’apprentissage est de construire un modèle à partir de données connues pour ensuite pouvoir l’appliquer à des instances inconnues. Dans le cadre de notre problème, nous souhaitons pour une équipe, au regard de ses décisions, être capable de l’intégrer dans un groupe de joueurs / classes défini lors du déploiement du clustering dans la partie précédente.

Principe de la régression logistique binaire

Dans une approche statistique, les décisions des équipes seront des variables notées X dans la suite du document. Le groupe ou la classe d’appartenance d’une équipe sera noté Y , sa classe. Y est donc la variable que nous cherchons à prédire avec les variables X .

On cherche à écrire l’espérance conditionnelle de la variable à expliquer (Y) comme combinaison linéaire de régresseurs X . Dans notre cas, nous souhaiterions classer une équipe en évaluant la somme pondérée de ses décisions.

L’objectif de la régression logistique est donc celui de toute régression : modéliser l’espérance conditionnelle $E(Y/X = x)$.

En régression linéaire, on cherche à faire passer un hyperplan au milieu du nuage des points $(x_1, x_2, \dots, x_p, y)$, de sorte que l’ensemble des valeurs moyennes de Y pour toutes les valeurs de X soit approché par cet hyperplan d’équation (TUFFÉRY [2012]) :

$$E(Y/X = x) = \beta_0 + \beta_1 X_1 + \dots + \beta_p X_p \quad (4.18)$$

avec β_k coefficient de la combinaison linéaire pour la variable explicative X_k .

Régression logistique : Le modèle *logit*

La fonction $f(p) = \log(p/1 - p)$ est appelée *logit*. Dans ce modèle, ce n'est plus l'espérance $\pi(x) = E(Y/X = x)$ qui est écrite comme combinaison linéaire mais $f(\pi(x))$. f est appelée fonction de lien (VERHULST [1845], HENSHER et GREENE [2003]).

Régression logistique multinomiale

On suppose que la variable à prédire Y prend m valeurs non ordonnées. Dans ce cas, le *logit* de $Prob(Y = 1/X = x)$ peut être remplacé par le *logit* généralisé (HAUSMAN et MC-FADDEN [1984]) :

$$\log \left(\frac{Prob(Y = j/X = x)}{Prob(Y = m/X = x)} \right) \quad (4.19)$$

Il est calculé par rapport à une modalité de référence (ici $Y = m$), et c'est cette expression, définie pour $1 \leq j \leq m$, qui est écrite comme combinaison linéaire des variables explicatives, donc les coefficients $(\beta_{i,j})_i$ varient pour chaque j .

Les probabilités s'écrivent :

— pour $j = 1, \dots, m - 1$:

$$Prob(Y = j/X = x) = \frac{\exp(\alpha_j + \sum_k \beta_{j,k} * x_k)}{1 + \sum_{i=1}^{m-1} \exp(\alpha_i + \sum_k \beta_{i,k} * x_k)} \quad (4.20)$$

— pour $j = m$:

$$Prob(Y = m/X = x) = \frac{1}{1 + \sum_{i=1}^{m-1} \exp(\alpha_i + \sum_k \beta_{i,k} * x_k)} \quad (4.21)$$

Arbre de décision

L'approche par arbre de décision est l'une des techniques de classification les plus intuitives et les plus répandues en datamining. Elle fournit des règles explicites de classement. Elle fait partie de la catégorie des classifications hiérarchiques descendantes supervisées.

La technique de l'arbre de décision est employée en classement pour détecter des critères permettant de répartir les individus d'une population en n classes prédéfinies. On commence par choisir la variable qui, par ses modalités, sépare le mieux les individus de chaque classe, de façon à avoir des sous-populations, que l'on appelle *nœuds*, contenant chacune la plus forte proportion possible d'individus d'une seule classe, puis on réitère la même opération sur chaque nouveau nœud obtenu jusqu'à ce que la séparation des individus ne soit plus possible ou souhaitable. Le choix de la variable de séparation est réalisé en tentant de maximiser le score d'une fonction.

Il existe de nombreuses méthodes de génération d'arbre de décision. La plupart du temps, celles-ci diffèrent par le choix de la fonction de score. Par exemple, l'*entropie* est exploitée par l'algorithme ID3. QUINLAN [2014] a amélioré ID3 pour donner naissance à l'algorithme C4.5 puis C5.0. L'indice *Gini* est, quant à lui, utilisé pour l'algorithme de CART (BREIMAN et collab. [1984]).

4.7.3 Phase 2 - Classification

Dans notre étude, nous cherchons à classer une nouvelle équipe qui est totalement inconnue. Comme expliqué précédemment, nous pouvons soit exploiter les classifieurs

construits en phase 1 soit appliquer directement une méthode de classification ne nécessitant pas de phase de construction de modèle. C'est dans ce contexte que nous présentons dans les prochains paragraphes deux méthodes que nous avons expérimentées.

Proximité au centroïd

Le principe de cette méthode est d'affecter un individu à la classe la plus proche (MANNING et collab. [2008]). Pour cela la distance au centroïd est utilisée. Ainsi la méthode calcule pour le nouvel individu x la distance $d_{x,i}$ qui le sépare du centroïd de la classe i . Nous affectons ensuite à l'individu x la classe \hat{y} qui correspondra à la distance $d_{x,i}$ minimale.

$$\hat{y} = \arg \min_{k \in C} d_{x,i} \quad (4.22)$$

Méthode des k plus proches voisins

Le principe général de la méthode *K-Nearest-Neighbor* (K-NN) est d'affecter un nouvel individu à une classe en fonction de la classe connue de ses k plus proches voisins (COVER et HART [1967]). La figure 4.14 illustre la méthode. Sur cette figure, l'individu à classer est représenté par un pentagone violet. Le classement se fait en fonction de l'occurrence d'apparition de la classe i parmi les k voisins présentés. Ainsi, dans notre exemple, si $k = 3$, nous classerons l'individu dans la classe 1 car parmi ses trois plus proches voisins, deux sont de la classe 1. Par contre, si l'on augmente la valeur de k à 5, cet individu sera classé dans la classe 2 car 3 voisins proches sont de cette classe alors que 2 proches voisins sont de la classe 1. Cet exemple illustre donc l'importance du choix du paramètre k .

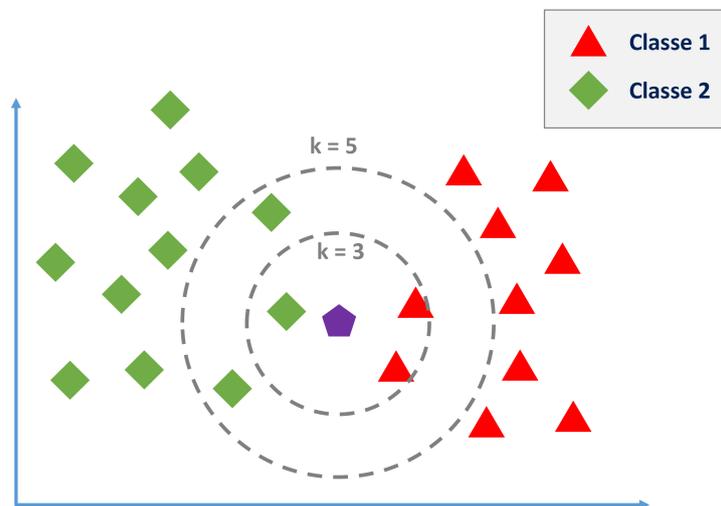


FIGURE 4.14 – Classification par K-NN

En général, k est impair car il permet d'obtenir une réponse unique lors de l'affectation lorsque 2 classes sont considérées.

Notons également que la méthode d'évaluation de la distance entre deux individus peut impacter le résultat. Pour nos expérimentations, nous avons souhaité comparer la distance calculée selon :

- la distance euclidienne classique donnée par la formule (4.23)
- une distance adaptée à notre contexte donnée par la formule (4.24)

Dans ce deuxième cas, la distance proposée est spécifique à notre contexte d'étude. Ainsi, l'écart a été pondéré par l'écart entre le cycle analysé sur l'historique (1 à n-1) et le cycle à prédire (n). Nous proposons de minimiser l'impact des cycles éloignés du cycle à prédire afin de donner plus d'importance aux derniers choix de l'individu.

Distance euclidienne :

$$D(X, Y) = \sqrt{\sum (X_j - Y_j)^2} \quad (4.23)$$

Distance pondérée proposée :

$$D(X, Y) = \sum 2^{-n} * (X_j - Y_j)^2 \quad (4.24)$$

n représente l'écart entre le cycle de la base de connaissances avec le cycle à prédire. Plus n est élevé et moins la distance sera impactante pour la classification.

Notons que ce choix s'oppose à l'exploitation de la distance euclidienne qui pondère toutes les distances antérieures avec un poids de 1. Elles ont toutes la même importance.

4.7.4 Évaluation de la performance de la classification

Avant de pouvoir classer de nouveaux individus, il est indispensable de soumettre la classification à une validation comme le présente la figure 4.15.

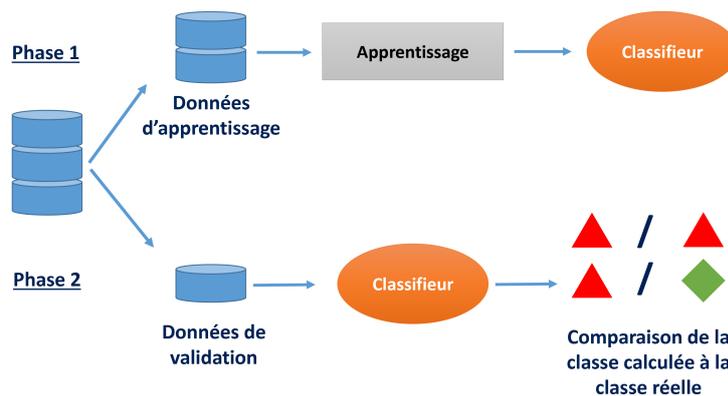


FIGURE 4.15 – Principe global de la validation de la classification

Le principe de la validation de la classification est de vérifier que la classification d'individus soumis à ce classifieur sera correcte. Pour réaliser cette opération, il convient de distinguer deux jeux de données. Le premier jeu sera exploité pour créer le classifieur. Ce seront les données d'apprentissage. Le second jeu de données constituera les données de validation. Tout comme dans le jeu de données d'apprentissage, les individus du jeu de validation ont une classe connue qui ne sera pas utilisée pour la classification, mais servira de contrôle. Il faudra donc affecter une classe à chaque individu du jeu de validation et comparer cette classe calculée à la classe réelle de l'individu. Cette comparaison sera agrégée dans une *table de contingence* qui présentera le taux d'individus du jeu de validation bien classés.

La constitution des jeux de données d'apprentissage et de validation peut se faire de différentes façons. Il est par exemple possible de scinder la base de données en deux de façon arbitraire (usuellement $\frac{2}{3}$ apprentissage et $\frac{1}{3}$ de validation).

Dans le cas de jeux de données à faible effectif, il est proposé de déployer une méthode de validation croisée *n-fold Cross Validation*. Cette fois, les jeux d'apprentissage et de test

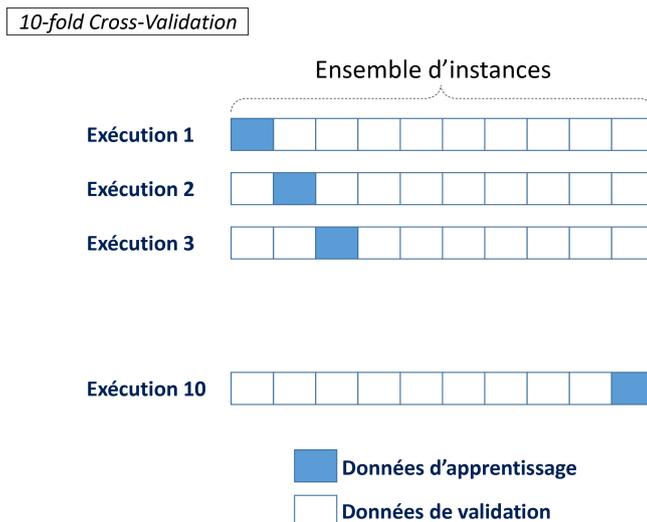


FIGURE 4.16 – Principe de la validation croisée

sont générés pour n exécutions. Lors de chaque exécution, $\frac{1}{n}$ sera la base de validation et $\frac{n-1}{n}$ sera le jeu d'apprentissage (figure 4.16).

Pour chacune des méthodes, le taux d'individus bien classés permet de valider ou d'invalidier le classifieur.

4.7.5 Classification sur les données de FirStrat

L'étape précédente de clustering a partitionné les équipes en différentes classes. Pour mémoire, nous avons choisi d'utiliser la fenêtre 5–8 et la création de 3 clusters. Nous souhaitons maintenant pouvoir affecter une nouvelle équipe à l'une des classes. Pour cela, nous devons déterminer la méthode de classification à utiliser.

Choix de la méthode de classification

Nous avons choisi de comparer différentes méthodes et d'évaluer leurs performances sur nos données. Nous avons donc scindé le jeu de données composé des décisions des équipes sur la fenêtre 5-8 en deux sous-ensembles. Un sous-ensemble sera réservé à l'apprentissage (2/3 des données) et l'autre à la validation. Nous avons ensuite dressé un tableau retraçant le taux d'équipes correctement classées par méthode déployée.

Le tableau 4.4 indique pour chacune des méthodes le taux d'individus bien classés. Les méthodes expérimentées sont :

- k plus proches voisins (distance euclidienne) avec différentes valeurs de k
- k plus proches voisins (distance pondérée) avec différentes valeurs de k
- Proximité du centroïd
- Régression logistique multinomiale (mlogit)

Le tableau 4.4 nous permet d'observer que le taux d'équipes biens classées varie entre 64,34% pour la régression logistique et 92,73% pour K-NN ($k = 7$). La méthode K-NN se démarque assez nettement. Quelque soit le mode de calcul de distance et la valeur de k , le taux d'équipes biens classées est supérieur à 88%. Il atteint 92,73% pour K-NN en distance pondérée et $k = 5$. Cette méthode est donc de bonne qualité. La régression logistique multinomiale et la proximité au centroïd sont moins adaptées.

TABLEAU 4.4 – Comparaison des méthodes de classification

Méthode	Taux de biens classés
K-NN / distance euclidienne	k = 3 88,63%
	k = 5 88,38%
	k = 7 90,61%
K-NN / distance pondérée	k = 3 91,48%
	k = 5 92,73%
	k = 7 90,92%
Centroïd	78,29%
mlogit	64,34%

Ces expérimentations nous permettent de conclure que, pour la suite des traitements, nous exploiterons la méthode K-NN avec $k = 5$ ainsi que la distance pondérée entre l'équipe à classer et le centroïd.

Classification sur données partielles

Un business game se déroule en tour par tour. Puisque nous souhaitons pouvoir classer les équipes en cours de partie, l'aspect temporel intervient dans la classification. Nous souhaitons effectuer des expérimentations sur nos données afin de nous assurer qu'il est possible de classer correctement une équipe (dans la bonne classe; classes qui ont été construites en utilisant les données sur des fenêtres des tours 5 à 8) avec des données partielles. Ainsi, nous allons calculer le taux d'équipes bien classées par la classification K-NN avec la distance pondérée, en ne disposant que des données des tours 2 à 3. Dans un second temps, nous ferons les mêmes expérimentations sur les données issues des tours 2 à 4. Nous répéterons ces essais afin de couvrir les différentes fenêtres de données. L'objectif est d'évaluer la variation de la qualité de la classification en fonction de la quantité de données connues représentant l'historique.

Nous pouvons être amenés à deux conclusions différentes.

- **Cas 1** : Il est impossible de classer correctement une équipe par exploitation partielle de ses variables. Dans ce cas, nous devons distinguer plusieurs partitionnements. Pour chaque période du jeu, il y a aura un nouveau partitionnement et donc une production plus importante de connaissances.
- **Cas 2** : Il est possible de classer rapidement une équipe par exploitation partielle de ses variables. Dans ce cas, un seul partitionnement suffit. Nous exploiterons uniquement les variables connues pendant le jeu pour réaliser une classification correcte.

Le tableau 4.5 indique pour chaque fenêtre de données utilisées pour la classification le taux d'équipes bien classées. Il a été obtenu en utilisant les classes découlant du clustering effectué sur la fenêtre 5-8 (Tableau 4.2). Nous pouvons observer qu'une augmentation de la quantité d'informations permet une amélioration de la classification. Cependant les valeurs les plus faibles observées sur ce tableau restent suffisantes pour notre contexte et au-delà de 70 % d'équipes bien classées.

Pour chaque fenêtre débutant à un tour donné, les meilleures performances sont indiquées en gras et confirment que l'augmentation de données entrantes améliore significativement la performance de classification.

Cette expérimentation permet de conclure que la classification au moyen de données partielles est correcte dans la plupart des cas. Cependant, il semble logique et pertinent

TABLEAU 4.5 – Taux d'équipes bien classées pour les variables sélectionnées

Données entrantes	Taux de bien classés
2-4	70,45 %
2-5	78,11 %
2-6	85,20 %
2-7	88,03 %
3-4	71,05 %
3-5	76,20 %
3-6	76,40 %
3-7	86,21 %
4-5	80,08 %
4-6	84,73 %
4-7	91,52 %
5-6	85,72 %
5-7	94,92 %

d'essayer d'exploiter un maximum de données pour la classification quand cela est possible. Cette étude nous montre que nous pouvons donc réaliser un seul clustering. Celui-ci permettra de caractériser les comportements des joueurs pour toute la durée d'une simulation.

Conclusions sur l'extraction de connaissances à partir des données de FirStrat

L'extraction de connaissances réalisée sur des données provenant d'un business game simple nous a permis de valider une démarche et de définir les méthodes à employer. En effet, nous sommes parvenus à réaliser un partitionnement pertinent des données présentées. La mise en œuvre de ce partitionnement nous a permis d'envisager plusieurs méthodes de classification et de sélectionner la meilleure.

Nous avons également vérifié la pérennité d'un partitionnement unique à exploiter lors de l'optimisation. Nous avons démontré qu'il était opportun de n'utiliser qu'un seul partitionnement pour toute la durée d'une partie.

Nous disposons donc désormais de la connaissance traitée et classée nous permettant de proposer une méthode originale de prédiction des données des joueurs. Nous présenterons dans la partie suivante cette méthode et déploierons des expérimentations sur les mêmes données issues de FirStrat.

4.8 Prédiction des décisions des équipes : Arbres d'anticipation

Nous disposons désormais des classes de joueurs. Depuis la section précédente de ce chapitre, nous sommes en mesure de positionner un joueur dans une classe et donc de connaître ou caractériser son comportement moyen. Nous souhaitons maintenant mettre en place, pour chaque classe, des arbres que nous nommerons *Arbres d'anticipation*. Ils nous permettront de suivre les décisions d'un joueur pour les tours de jeu passés et d'anticiper les décisions pour les tours suivants.

Nous nous positionnons donc naturellement dans la dernière partie du schéma 4.17. Ces arbres seront évidemment exploités avant le processus d'optimisation présenté au chapitre 3. Ils fourniront des informations fondamentales au processus d'optimisation.

Ces informations permettront de prédire l'impact des décisions des équipes sur le marché et donc d'évaluer les ventes prévisionnelles à intégrer à l'optimisation.

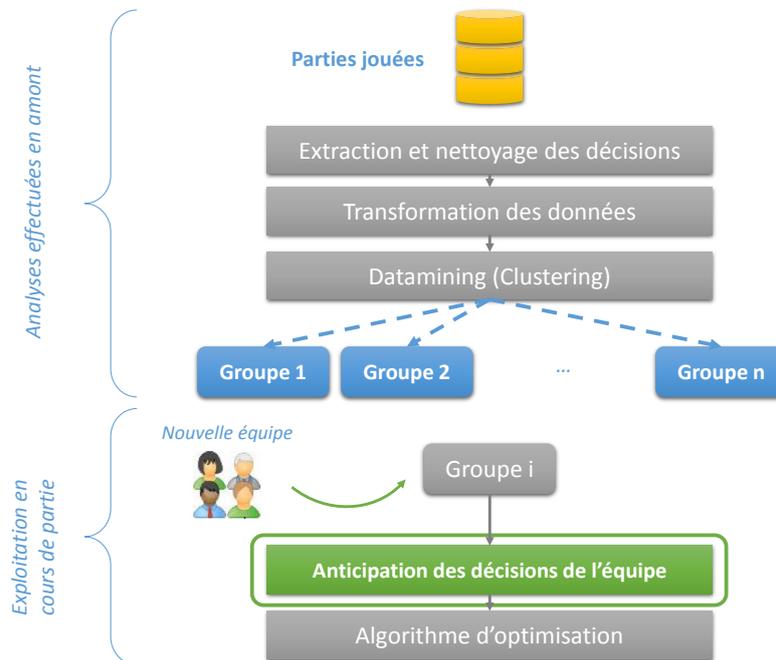


FIGURE 4.17 – Méthodologie générale

Nous avons souhaité nous inspirer de la méthodologie de l'arbre de décision de QUINLAN [2014] et de LOH [2011]. Contrairement aux arbres de décisions, nous ne cherchons pas à classer un individu. Ceci est déjà effectué. Nous souhaitons améliorer notre connaissance de chaque classe en identifiant le comportement majoritaire au sein de celle-ci.

4.8.1 Les arbres d'anticipation

Méthode de construction des arbres d'anticipation

Nous souhaitons construire un outil nous permettant de suivre les décisions d'un joueur afin d'être en mesure de les anticiper. Ce suivi de décisions sera réalisé au sein d'une classe (le joueur aura déjà été classé). Nous souhaitons donc retracer son parcours au sein de la classe en analysant son historique. Une fois ce parcours connu, nous cherchons à anticiper une décision cohérente pour lui à l'étape suivante. Cette décision serait exploitable en mode *coach* comme en mode *concurrent*.

La construction de l'arbre suit un processus itératif comme présenté sur la figure 4.18. L'initialisation de la construction de l'arbre se fait par la sélection d'une variable. Puisque nous souhaitons analyser les décisions des premiers tours, pour anticiper les décisions des tours suivants, l'aspect temporel des variables est très important. Les variables seront donc considérées tour après tour (tour n , puis $n + 1 \dots$). La variable sélectionnée correspondra au tour de jeu le plus petit possible. Il s'agira donc du premier mois de la fenêtre sélectionnée. Le mode de sélection de la variable sera discuté par la suite. Une fois cette variable sélectionnée et le branchement réalisé, une nouvelle étape de sélection de variable est lancée. La nouvelle variable sélectionnée ne pourra pas représenter un tour de jeu antérieur à la variable précédente. L'arbre ne pourra qu'être parcouru dans une direction.

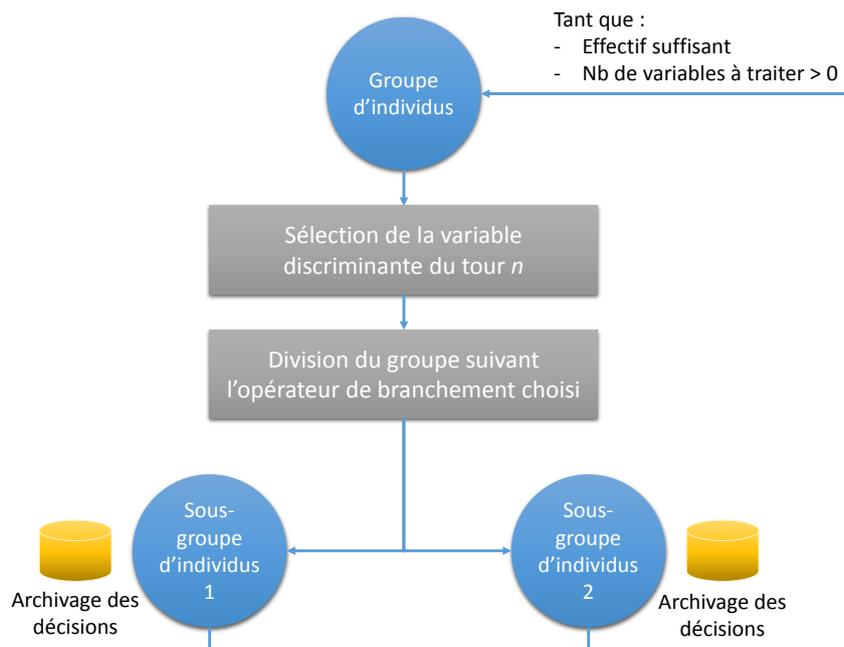


FIGURE 4.18 – Principe de l'arbre d'anticipation

Nous avons cherché à identifier le meilleur moyen de sélectionner la variable la plus appropriée à notre problème. Ainsi, nous proposons d'étudier :

- La variable dont la variance est la plus faible (pour identifier des caractéristiques communes au sein des groupes)
- La variable dont la variance est la plus forte (pour identifier des différenciations au sein des groupes)

Cet algorithme dispose d'une condition arrêt intrinsèque : le traitement de toutes les variables décisionnelles. Cependant, nous avons souhaité ajouter un critère d'arrêt correspondant au cardinal du groupe d'individus restant à traiter. Si nous voulons maintenir un bon niveau de prédiction, nous devons nous assurer que les groupes et sous-groupes créés disposent d'un nombre d'individus suffisant pour proposer une projection fiable.

L'étape de branchement du groupe d'individus en deux sous-groupes nous impose de définir la limite de séparation. Une fois les groupes séparés, nous stockons pour chaque nœud les statistiques univariées des variables de la branche. Ces statistiques permettent de définir lors du parcours de l'arbre les décisions communément prises et exploitées pour le traitement d'optimisation.

Pour l'opérateur de branchement, nous pouvons opter pour d'une part **la médiane** et d'autre part la **moyenne encadrée** par une proportion de son écart-type. La médiane nous assurera une répartition en groupes homogènes en nombre d'individus. Nous pourrions donc certainement construire des arbres disposant de plus de branches que par l'autre opérateur de branchement proposé. En effet, la moyenne encadrée nous permettra de sélectionner les comportements les plus couramment observés au risque de déséquilibrer l'arbre. Le risque de cette deuxième option est bien de réduire trop rapidement l'effectif des groupes minoritaires et ainsi de stopper très vite la construction de l'arbre ce qui aurait comme conséquence d'interdire une projection correcte des décisions concurrentielles.

Exemple d'arbre

Nous obtenons alors pour chaque classe un arbre similaire à la figure 4.19 qui présente une partie d'un arbre.

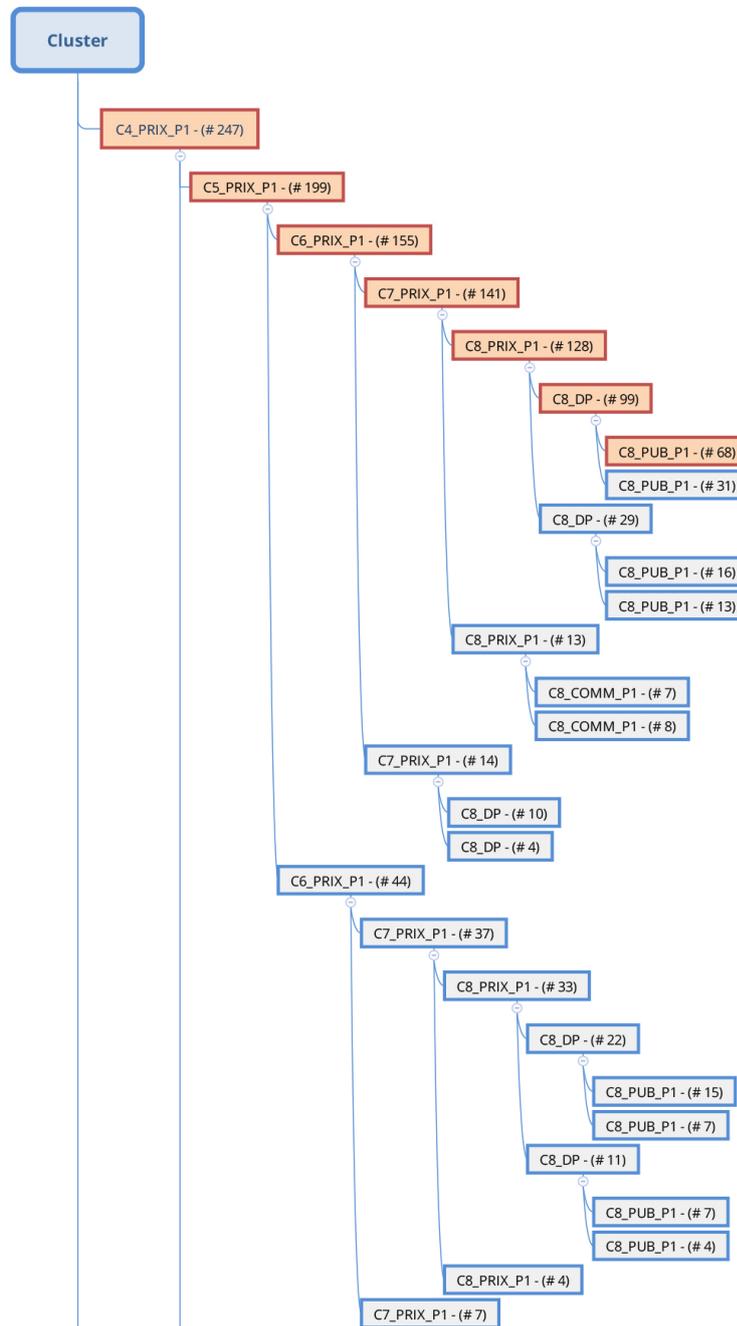


FIGURE 4.19 – Exemple d'arbre d'anticipation

La représentation graphique et partielle de l'arbre ci-dessus présente sur chaque branche le nom de la variable discriminante. L'effectif de chaque groupe figure entre les parenthèses.

Dans l'exemple présenté, à titre indicatif, nous avons choisi de couper l'effectif par la moyenne encadrée d'un écart-type (noté σ). La variable discriminante a été la variable présentant la plus faible variance.

Les équipes comprises dans l'intervalle $[\bar{x} - \sigma, \bar{x} + \sigma]$ sont toujours situées sur la branche supérieure. Les autres occupent la branche inférieure.

Ainsi, lorsque nous parcourons l'arbre en suivant la branche supérieure (cellules de couleur rouge-brique), nous disposons des décisions les plus fréquemment prises (autour de la moyenne $+/-\sigma$).

4.8.2 Protocole d'expérimentation des arbres d'anticipation

Afin d'évaluer la pertinence de l'algorithme, nous avons établi un protocole de test. Nous avons exécuté l'algorithme sur les 90 parties déjà jouées de la base de connaissances du simulateur FirStrat. Ces parties regroupent 590 équipes. Nous avons souhaité prédire la décision du tour n en nous basant sur les décisions 1 à $n - 1$. Le parcours des arbres ne devient plus pertinent pour n trop faible car l'historique de l'équipe fait défaut.

Cette série d'expérimentations nous permet de tester :

- Le mode de construction de l'arbre.
- La pertinence des données extraites de l'arbre.

4.8.3 Validation de la construction de l'arbre

La construction des arbres d'anticipation nous demande de nous interroger sur les critères suivants :

- Choix de la variable discriminante qui définira le branchement de l'arbre.
- Choix du groupage des individus sur chaque branche de l'arbre.

Sélection de la variable

Nous proposons de réaliser nos expérimentations en sélectionnant pour variable discriminante :

- la variable qui présente la plus grande variance,
- la variable qui présente la plus petite variance.

Opérateur de branchement

Le choix de l'opérateur de branchement n'est ni intuitif ni anodin.

Afin d'illustrer les conséquences de ce choix, nous avons réalisé la figure 4.20. Cette figure présente pour chaque étape de construction (mentionnée à gauche), deux arbres générés. L'arbre de gauche a été construit en choisissant la médiane comme opérateur de branchement. L'arbre de droite a été réalisé avec la moyenne encadrée pour opérateur de branchement. Dans chaque bulle, nous avons indiqué à titre d'exemple les effectifs des sous-groupes. La génération de l'arbre est stoppée dans notre figure par un effectif trop faible dans un sous-groupe.

Nous observons que l'arbre construit en choisissant la médiane comme opérateur de branchement permet d'obtenir des arbres équilibrés. Chaque sous-groupe compte le même effectif. Dans l'autre cas, les branches situées à droite de l'arbre comptent toujours plus d'équipes que les branches de gauche. Les arbres sont donc déséquilibrés. Les branches de droite permettent aussi naturellement d'observer le comportement « général » de la classe.

Dans le cas de la moyenne, la construction itérative sera stoppée plus rapidement lors des itérations par manque d'équipe dans le dernier sous-groupe réalisé. Si nous optons

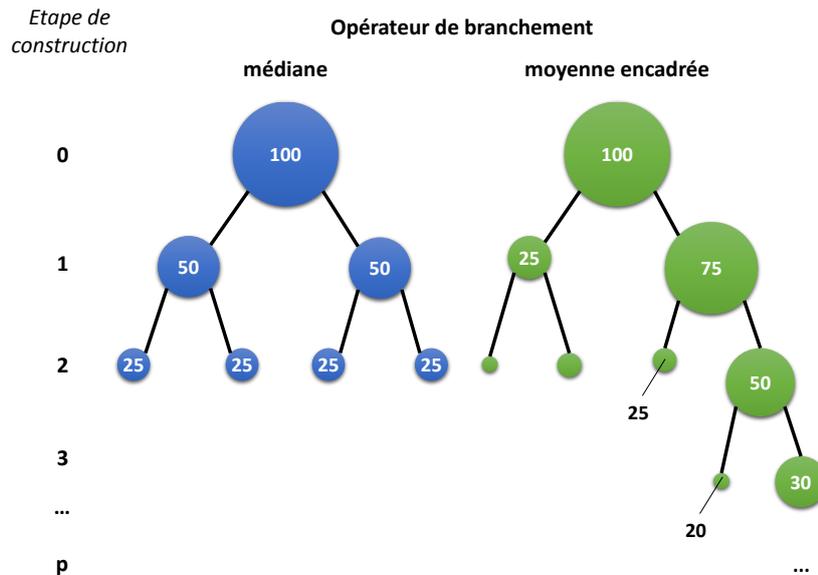


FIGURE 4.20 – Représentation de la morphologie des arbres en fonction du choix de l'opérateur de branchement

pour la moyenne encadrée, la construction de l'arbre suivra plus d'étapes et nous augmenterons donc le nombre de variables sélectionnées.

Afin de déterminer la meilleure stratégie de construction des arbres, nous avons opté, pour une première série de tests, pour un branchement en sélectionnant les équipes dont la valeur accordée à la variable discriminante préalablement choisie est comprise dans un intervalle encadrant la moyenne par une fraction de l'écart-type.

Ce choix impose nécessairement un arbre déséquilibré volontairement. Nous privilégions la norme représentée par la moyenne.

Dans un autre test, nous avons souhaité équilibrer les groupes de l'arbre et ainsi avons divisé les groupes par rapport à la médiane du groupe. Le résultat produit donc des arbres parfaitement équilibrés.

Ainsi, nous comparons quatre configurations de construction de l'arbre :

1. Variance minimale + Moyenne encadrée
2. Variance minimale + Médiane
3. Variance maximale + Moyenne encadrée
4. Variance maximale + Médiane

4.8.4 Expérimentations

Les expérimentations ont été menées pour 590 individus et pour 4 instances ou paramètres. Les 590 individus représentent des équipes prises dans 90 parties différentes. Pour chacune de ces équipes, nous avons considéré les tours de jeu allant de 3 à 8.

Définition d'un paramétrage correct

Afin de valider le paramétrage des arbres d'anticipation et surtout le définir, nous avons comparé trois méthodes pour prédire les décisions de chaque équipe. L'idée étant d'évaluer si les arbres d'anticipation permettent de mieux prédire le comportement des

équipes. Ces prédictions ont été réalisées pour chaque équipe et pour chaque tour compris entre les tours 3 et 8 à partir :

- de la moyenne générale du jeu de données,
- du centroïd de la classe affectée à l'équipe,
- de l'arbre d'anticipation pour les 4 configurations données dans la section précédente.

La figure 4.21 nous permet de visualiser à quelle étape de l'approche globale, interviendraient ces prédictions. La première méthode provient directement et sans retraitement de l'historique des simulations. La seconde méthode est le fruit du clustering. Les deux premières méthodes ne nécessitent ainsi aucune phase d'apprentissage. Elles sont calculés a priori donc sans traitement pendant l'exploitation. La dernière méthode (et ses 4 configurations) est issue de l'apprentissage.

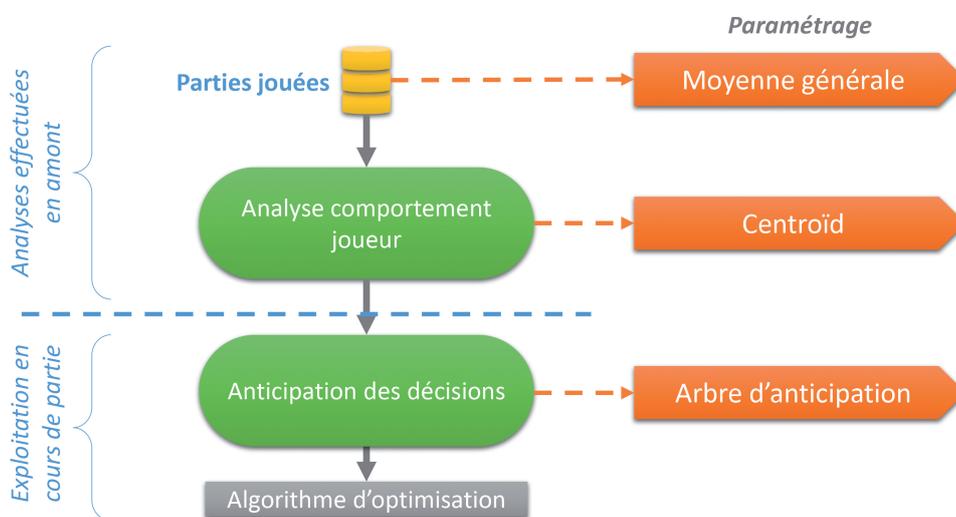


FIGURE 4.21 – Provenance des paramètres dans le processus d'optimisation

A l'issue de ces expérimentations, nous disposons donc de 6 ensembles de décisions prédites (un par méthode sans apprentissage et quatre pour la méthode des arbres d'anticipation) que nous voulons comparer à la décision réelle de l'équipe. Nous avons alors calculé la distance euclidienne entre la décision réelle du joueur et la décision prédite par les différentes méthodes. Nous disposons donc des tableaux de distances comparées.

Les tableaux 4.6, 4.7 et 4.8 présentent les comparaisons de 2 méthodes deux à deux. Ils indiquent le nombre de fois où l'une des méthodes est plus proche de la décision réelle que l'autre. Ces résultats nous permettent donc de sélectionner, parmi ces deux méthodes, celle qui offre la décision la plus proche de la décision réelle du joueur le plus fréquemment.

Sur le tableau 4.6, nous observons que la distance offerte par l'exploitation du centroïd est dans 2 cas sur 3 plus proche de la décision de l'équipe réelle que la moyenne générale.

Nous avons poursuivi nos expérimentations en comparant la méthode centroïd à la méthode des arbres. Chaque configuration de l'arbre a été comparée à la méthode du centroïd. Le tableau 4.7 présente les taux pour lesquels chaque paramétrage s'est classé en première position (par rapport à la méthode du centroïd).

Parmi les quatre paramétrages de la méthode des arbres d'anticipation, le paramétrage consistant à prendre l'arbre généré en sélectionnant la variable qui présente la variance la plus faible et en séparant les individus par rapport à la moyenne encadrée semble

TABLEAU 4.6 – Comparaison des paramètres : centroïd vs. moyenne générale

Décision	Centroïd vs. Moyenne
Moyenne	64%
Tour 3	65%
Tour 4	66%
Tour 5	65%
Tour 6	64%
Tour 7	64%
Tour 8	62%

TABLEAU 4.7 – Comparaison des paramètres des arbres par rapport à la méthodes des centroïds

Sélection de variable Opérateur de branchement	<i>min</i> (σ)		<i>max</i> (σ)	
	Moyenne	Médiane	Moyenne	Médiane
Moyenne	69%	68%	68%	66%
Tour 3	63%	60%	63%	61%
Tour 4	64%	64%	61%	60%
Tour 5	71%	68%	70%	68%
Tour 6	76%	73%	74%	75%
Tour 7	72%	71%	70%	72%
Tour 8	74%	70%	71%	58%

se distinguer. En effet, la distance entre le centroïd de la classe et la valeur prédite par l'arbre généré dans ces conditions est toujours mieux classée que les autres paramétrages.

Enfin, nous avons souhaité évaluer le bénéfice global des arbres d'anticipation par rapport à la moyenne générale des décisions issue de l'historique. Nous souhaitons également nous prononcer sur le cycle à partir duquel l'exploitation des connaissances apportent une plus-value par rapport à la moyenne générale. En effet, serait-il plus judicieux d'exploiter simplement la moyenne générale pour les premiers tours de jeu? L'intégration rapide des arbres nous permettra-t-elle d'améliorer rapidement la prédiction des décisions des équipes?

Pour répondre à ces questions, nous avons, pour les tours 3 à 8, comparé la distance entre la décision des joueurs et la décision prédite d'une part par la moyenne générale et d'autre part par les arbres générés par le meilleur paramétrage. Le tableau 4.8 fait état des pourcentages pour lesquels la décision émanant de l'arbre a été plus proche de la décision issue de la moyenne générale.

TABLEAU 4.8 – Comparaison des paramètres des arbres par rapport à la moyenne générale

Décision	Taux d'amélioration
Moyenne	87 %
Tour 3	83 %
Tour 4	83 %
Tour 5	85 %
Tour 6	88 %
Tour 7	90 %
Tour 8	91 %

Ce dernier tableau confirme que nous sommes parvenus dans plus de 80 % des cas, et ce dès le tour 3, à réduire la distance entre la décision prédite et la décision réelle au moyen de l'exploitation des arbres suivant le paramétrage :

- Sélection de variable = Variance minimale
- Branchement = Moyenne encadrée

La méthode proposée dans le cas du simulateur FirStrat consistant à l'élaboration d'arbre d'anticipation permet d'améliorer fortement la qualité de la décision prédite. Ainsi, nous préconisons, pour les tours 1 et 2 d'exploiter la moyenne générale. L'extraction de connaissances n'était pas possible en début de jeu. Dès le tour 3, les décisions produites par les arbres apportent une réelle amélioration de la prédiction et s'imposent à la moyenne générale.

4.9 Conclusions

Dans ce chapitre, nous avons souhaité analyser le comportement des joueurs afin d'anticiper au mieux leurs décisions et ainsi d'en évaluer leur impact sur le marché. Pour atteindre cet objectif ambitieux, nous nous sommes focalisés sur la conception et la validation de la démarche que nous avons dévoilée au fil de ce chapitre. Ce souhait nous a amenés à écarter, pour un temps, le simulateur UpStrat présenté au chapitre 1. Nous avons préféré nous tourner vers FirStrat, simulateur aux caractéristiques plus modestes. Ce dernier nous a permis de nous concentrer sur la méthodologie générale.

Ainsi, nous avons dans un premier temps mis en lumière ce nouveau simulateur et les données produites par celui-ci. Nous avons aussi présenté les données extraites et les pré-traitements que nous avons effectués. Dans un second temps, nous avons appliqué des méthodes de clustering aux données et avons cherché à en évaluer leur performance. Les résultats de cette opération nous ont offert l'opportunité de comparer le comportement des joueurs au modèle stratégique de M. Porter. Nous nous sommes inquiétés de la stabilité du clustering choisi au cours des parties. Ensuite, la comparaison des méthodes de classification nous a conduits au choix de la méthode la plus appropriée pour notre étude.

La suite de la démarche concernait les traitements à effectuer en phase d'exploitation. Cette étape s'est concentrée sur l'exploitation des connaissances produites précédemment. Nous avons ainsi présenté un modèle d'arbre d'anticipation. Lors des expérimentations menées en fin de chapitre, nous avons remarqué que l'exploitation de ces arbres permettait à la décision prédite de s'approcher de la décision réelle des joueurs. Nous avons en conclusion préconisé d'exploiter en début de partie la moyenne générale et dès le tour 3 de faire confiance aux arbres correctement paramétrés.

Dans le chapitre suivant, nous revenons au simulateur UpStrat. Nous voulons confronter la démarche proposée dans ce chapitre aux modalités du simulateur UpStrat. Cette nouvelle situation nous mènera, nous l'espérons, vers la proposition d'une approche générique de nos travaux.

Ce dernier chapitre accueille également les résultats des expérimentations issues de l'exploitation des arbres lors du processus d'optimisation. Nous comparons donc le classement obtenu par le processus d'optimisation complet avec et sans exploitation de connaissances.

Chapitre 5

Évaluation de l'approche sur UpStrat

Dans le premier chapitre, nous nous sommes concentrés sur le contexte de notre étude et sur la présentation des business games en général et du simulateur UpStrat en particulier. Nous avons ensuite, dans le chapitre 2, proposé une modélisation du cas d'étude en un problème d'optimisation combinatoire. Celui-ci a été décomposé en plusieurs éléments et nous avons présenté une modélisation sous forme de programme linéaire pour les départements Production, Distribution et Finance des entreprises virtuelles. Dans le chapitre 3, nous avons présenté une méthode de recherche locale qui, couplée aux programmes linéaires appliqués à chaque département, offrait des décisions économiques performantes dans un contexte économique donné. Ainsi, à la fin du chapitre 3, nous disposons d'agents capables de prendre des décisions cohérentes sans réaction vis à vis de l'environnement économique.

C'est pourquoi, dans le chapitre 4, nous nous sommes concentrés sur les réactions du marché par rapport à l'agent et ses concurrents. Pour cela, il était nécessaire de pouvoir anticiper le comportement des concurrents. Nous avons donc proposé une étude de ces comportements qui nous a conduits à la construction d'arbres d'anticipation. Dans ce même chapitre, afin de valider nos propositions, nous nous sommes écartés du simulateur initialement choisi, UpStrat, au profit du simulateur FirStrat qui présente des données plus simples à traiter et à analyser. Les résultats obtenus nous ont permis d'envisager une transposition de la méthode d'anticipation des comportements de concurrents au simulateur UpStrat, cas initial de l'étude.

Nous nous trouvons dans la phase d'analyse de la réaction concurrentielle comme indiqué sur la figure 5.1.

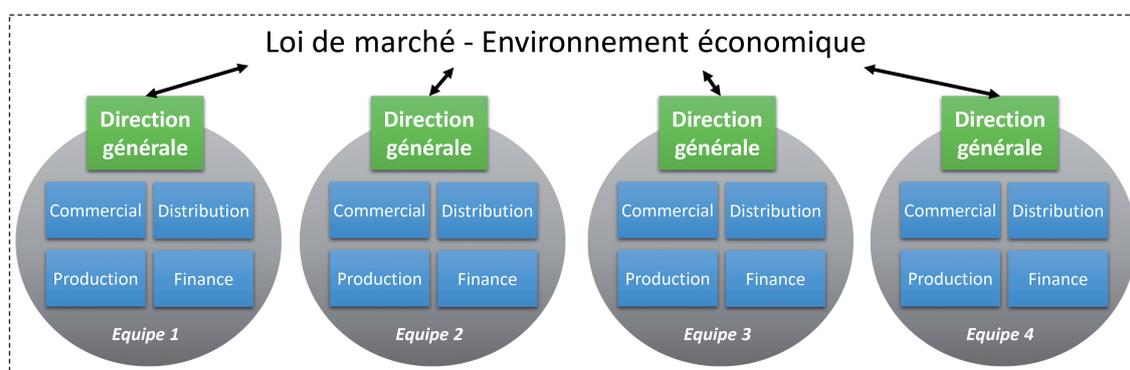


FIGURE 5.1 – Environnement d'une simulation

Ainsi, l'objet de ce chapitre est de valider la démarche globale en intégrant toutes les optimisations et résultats obtenus dans les chapitres précédents. Nous débuterons ainsi

par la phase d'extraction de connaissances qui nous conduira, par la suite, à la construction des arbres d'anticipation sur les données de UpStrat. Nous procéderons de manière analogue à ce que nous avons fait sur FirStrat. Nous évaluerons la pertinence d'intégrer ces arbres à l'optimisation globale. Nous nous intéresserons en particulier aux méthodes à employer et au moment privilégié pour leur emploi pendant une session de jeu.

Enfin, nous évaluerons le processus de prise de décision complet sur un ensemble de parties déjà jouées. Ainsi, nous comparerons les performances de nos agents à la performance des équipes réelles. Aussi, il nous paraît important d'évaluer la contribution de l'apport de connaissances dans l'anticipation des décisions des joueurs et donc sur la performance globale de l'optimisation. Nous procéderons donc à une comparaison de nos expérimentations en envisageant une configuration sans apport de connaissances et une autre qui bénéficiera de cet apport.

En conclusion de ce chapitre, nous proposerons de nouvelles pistes de recherche et envisagerons l'exploitation opérationnelle des agents.

5.1 Extraction de connaissances sur UpStrat

Dans cette section, nous allons déployer le processus d'extraction de connaissances proposé précédemment aux données UpStrat. Nous allons donc dans un premier temps présenter les données puis dans un deuxième temps réaliser un clustering pour identifier les types de joueurs. Nous construirons dans un troisième temps les arbres d'anticipation pour chaque classe issue du clustering et réaliserons les tests et validations nécessaires pour ce simulateur.

Afin de valider l'aspect générique des méthodes proposées précédemment pour la partie extraction de connaissances, nous avons choisi, à nouveau, de comparer les différents indices de qualité, méthodes de clustering... Ceci nous permettra de nous assurer que nous obtenons les mêmes conclusions.

5.1.1 Présentation des données

Les données utilisées pour évaluer la performance de l'approche globale ont été recueillies sur environ 60 parties qui se sont déroulées sur des périodes de 2,5 à 3,5 jours. Chaque partie compte de 5 à 10 équipes. Chaque équipe est composée de 5 participants. Les 1 750 participants sont ainsi répartis en 350 équipes composées d'étudiants d'écoles de management et d'écoles d'ingénieurs. Les parties choisies sont jouées sur au moins 6 tours de jeu.

Les données recueillies sont très hétérogènes sur les aspects concurrentiels (nombre de concurrents) et temporels (nombre de tours de jeu réalisés). Notons que l'intensité concurrentielle au sein d'un univers augmente très fortement en fonction du nombre de concurrents. Les équipes sont ainsi amenées à prendre plus de risques et à accroître leurs investissements commerciaux. L'intensité concurrentielle impacte le comportement des équipes.

Cependant, tous les jeux se sont déroulés sur le même marché économique et dans les mêmes conditions d'animation. Chaque équipe a développé son activité sur le marché du cycle sur 3 produits que nous nommerons P1, P2 et P3. Nous retrouvons sur ce marché différentes gammes : un produit P1 « lowcost » avec des grands volumes mais de faibles marges, un produit P2 de milieu de gamme sur lequel une combinaison fine de facteurs clés de succès est exigée et enfin un produit P3 « luxe » ou « premium » vendu sur

un marché confidentiel mais assez rapidement profitable. Le produit P3 n'est pas immédiatement disponible. Les premières ventes arrivent au tour 3. Lors du tour 2, les équipes peuvent réaliser des investissements technologiques afin d'anticiper son lancement commercial sur le marché. Le scénario se déroule sur 3 zones distinctes. Les deux premières zones sont disponibles en début de jeu et sont matures. La troisième zone offre un potentiel de développement sur un marché naissant. Cette zone est disponible à partir du tour 4.

Un prétraitement a été effectué sur les données pour, d'une part, regrouper toutes les décisions, et d'autre part, normaliser et sélectionner les données. La base de données constituée présente 170 variables de décision.

5.1.2 Segmentation des données par produit

Au chapitre 2, nous avons présenté une méthode générique d'analyse stratégique et une méthode de comptabilité de gestion connue sous le nom de « coût complet ». Ces méthodes nous ont permis de valider la décomposition de l'entreprise en un ensemble d'activités autonomes regroupées sous le nom de Domaines d'Activités Stratégiques (DAS). Nous souhaitons appliquer ces méthodes à notre jeu de données afin :

- de réduire le nombre de variables de décisions à considérer,
- d'améliorer la caractérisation des équipes de joueurs.

Nous proposons d'identifier trois DAS par entreprise. Chaque DAS portera le nom d'un produit. Cette opération semble pertinente car la modélisation du marché offerte par UpStrat définit pour chaque couple produit/marché un ensemble de facteurs clés de succès différent et dont l'évolution est aussi différente. Cela signifie qu'un positionnement spécifique peut être appliqué à chaque couple produit/marché.

De plus, nous dénombrons peu de décisions concurrentielles communes aux trois produits. Le délai de paiement par zone fait partie de cette liste. Il est optimisé dans le département finance. La capacité de distribution est elle aussi commune aux trois produits. Elle est optimisée dans le département distribution. L'importance accordée à ces décisions pour le jeu de données à analyser est assez faible. Elle est fixée dans le scénario. Les efforts tarifaires, publicitaires, technologiques et de différenciation sont bien plus importants.

Nous considérerons donc que nos produits sont **indépendants** et que la caractérisation des comportements peut se faire, elle aussi, indépendamment. Ainsi, pour chaque produit et pour chaque cycle, les variables décisionnelles analysées seront le prix (par marché) ou la publicité (par marché) ou les options (par marché) ou encore les efforts technologiques.

Le nombre de variables à considérer évolue à chaque cycle pour chaque produit. Cette évolution est principalement la conséquence de l'ouverture de nouveaux marchés. Les investissements consentis en recherche et développement augmentent aussi les alternatives offertes aux équipes. Ces derniers permettent des positionnements industriels (production ou sous-traitance) et offrent des possibilités de différenciation. Les premiers cycles donnent un nombre faible de variables (5). En fin de simulation, nous dénombrons au plus 10 variables par tour de jeu et par produit.

5.2 Analyse des comportements

L'analyse des comportements des joueurs sera utile pour anticiper leurs décisions. Ces anticipations nous permettront d'estimer les ventes prévisionnelles de notre agent décisionnel. Ces ventes décisionnelles constitueront les données d'entrée du process d'optimisation.

La figure 5.2 nous permet de rappeler que ces ventes prévisionnelles seront exploitées en entrée de la phase d'optimisation. L'analyse des comportements des joueurs est, pour sa part, réalisée en amont.

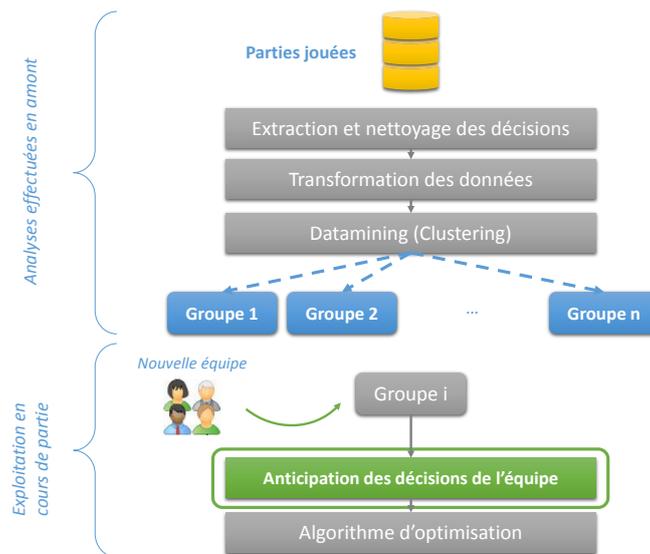


FIGURE 5.2 – Méthodologie générale

Nous allons, dans cette section, transposer la méthodologie du traitement appliqué dans le chapitre 4. Nous validerons également sa pertinence.

5.2.1 Clustering

Afin de valider la généralisation des conclusions obtenues sur les données du simulateur FirStrat, nous avons effectué sur les données UpStrat les mêmes expérimentations. Ainsi, nous avons réalisé un clustering suivant une méthode hiérarchique (D.Ward) et un autre suivant une méthode de partitionnement (K.means). Nous avons ensuite cherché à valider le clustering et à identifier le nombre de clusters offrant le meilleur indice de pertinence suivant les index Calinski-Harabasz (CH) et Davies-Bouldin (DB). Pour rappel, l'index CH doit être maximisé et l'index DB doit être minimisé.

Le tableau 5.1 reprend dans chaque cellule et pour chaque méthode employée le nombre de clusters offrant le meilleur indice de performance pour une fenêtre de données spécifiée sur les données sélectionnées. A titre d'exemple, nous lisons dans la première cellule sur la première ligne du tableau « 3 82,206 ». Cela signifie que le meilleur indice CH (82,206) a été obtenu pour un clustering sur 3 clusters pour le produit P1, sur la fenêtre de données 2_4 et pour la méthode de clustering K.means. Les traitements ont été effectués sur les fenêtres de données indiquées dans la deuxième colonne du tableau. La méthode K.means a été lancée une fois.

Nous avons souligné pour chaque produit et pour chaque couple index/méthode de clustering les meilleurs résultats obtenus (indice de performance) en fonction de la fe-

nêtre de décisions choisie. Nous avons également mis en gras la fenêtre de données sélectionnée pour la suite de nos traitements.

TABLEAU 5.1 – Méthodes de clustering et pertinence pour UpStrat par produit

		CH index (max)				DB index (min)			
		# clust.	Kmeans	# clust.	Ward.D2	# clust.	Kmeans	# clust.	Ward.D2
P1	<u>2_4</u>	<u>3</u>	<u>82,2026</u>	<u>3</u>	<u>66,9001</u>	<u>4</u>	<u>1,5503</u>	<u>5</u>	<u>1,4732</u>
	2_5	3	68,7975	3	52,5337	4	1,7885	3	1,7422
	2_6	3	60,8407	3	50,2855	5	1,9251	4	1,8351
	3_5	3	66,5659	3	58,7196	5	1,7236	4	1,7199
	3_6	3	58,1975	3	46,3574	5	1,9061	4	1,7997
	4_6	3	66,5651	5	49,2188	3	1,6165	3	1,7351
P2	<u>2_4</u>	<u>3</u>	<u>84,2377</u>	<u>3</u>	<u>71,5411</u>	<u>4</u>	<u>1,4742</u>	<u>3</u>	<u>1,5728</u>
	2_5	3	76,9866	3	64,8357	4	1,7138	3	1,9041
	2_6	3	75,0329	3	59,5575	4	1,8868	4	1,8506
	3_5	3	74,7023	3	62,5882	4	1,6658	4	1,7231
	3_6	3	68,7507	3	61,0214	4	1,8815	5	1,9888
	4_6	3	82,7271	<u>3</u>	<u>72,4821</u>	3	1,8102	3	1,9155
P3	3_5	3	91,3766	3	78,5926	<u>3</u>	<u>1,7357</u>	5	1,8182
	3_6	3	93,9103	<u>3</u>	<u>84,4368</u>	4	1,9224	3	1,9389
	<u>4_6</u>	<u>3</u>	<u>106,3371</u>	<u>3</u>	<u>83,041</u>	<u>3</u>	<u>1,7419</u>	<u>3</u>	<u>1,7005</u>

Pour chaque produit, noté ici P1, P2 et P3, les meilleurs indices de performance sont obtenus pour la méthode de clustering par partitionnement (K.means). Cette méthode est donc sélectionnée.

Les quatre meilleurs indices de performances pour le produit P1 sont obtenus pour la fenêtre de données 2_4. Elle est donc validée. Le nombre optimal de clusters varie entre 3 pour l'indice CH et 4 pour l'indice DB. Nous choisirons 3 clusters afin de maximiser les effectifs de chaque cluster et de rendre lisible le comportement moyen des individus de chaque cluster.

Le produit P2 présente des conclusions similaires au produit P1. En effet, la méthode K.means semble être la plus performante. La fenêtre de données est aussi la même : 2_4. Notons que la fenêtre 4_6 est préconisée par la méthode D.Ward. Cependant, l'écart entre les deux indices de performance (71,5411 et 72,5821) est faible et n'entraîne pas de modification de nos conclusions. Enfin, pour les mêmes raisons que pour le produit 1, nous garderons 3 clusters.

Le dernier produit présente moins de fenêtres de décisions que les autres produits (3 contre 6) car il arrive tardivement sur le marché concurrentiel. Concernant le choix de la méthode de partitionnement, pour l'indice CH, la méthode K.means est très nettement supérieure à la méthode D.Ward. La conclusion est inversée pour l'indice DB. Cependant, l'écart de performance entre les deux valeurs obtenues pour l'indice DB (1,7419 contre 1,7005) nous permet de choisir définitivement la méthode K.means. Pour l'ensemble des méthodes, le nombre de trois clusters revient systématiquement. Nous validons ainsi cette valeur. La fenêtre de décision peut être discutée. En effet, la fenêtre 4_6 pour K.means/CH (106,3371) est préférable alors que pour K.means/DB (1,7357) la fenêtre 3_5 est la plus appropriée. Sur ce dernier cas, nous remarquons que la différence entre les indices obtenus pour les deux fenêtres 4_6 et 3_5 est très faible. Nous observons 1,7357 et 1,7419. Ainsi, nous pouvons sans risque majeur valider la fenêtre 4_6.

Il paraît utile de synthétiser ces conclusions. Ainsi, le tableau 5.2 reprend les clusters sélectionnés pour la suite de l'étude.

TABEAU 5.2 – Caractérisation des méthodes de clustering pour chaque produit

	P1	P2	P3
Méthode	Kmeans	Kmeans	Kmeans
Fenêtre	2-4	2-4	4-6
Nombre de clusters	3	3	3

5.2.2 Description des classes de joueurs

De la même manière que nous l'avons fait sur FirStrat, nous pensons qu'il est judicieux de caractériser les joueurs de chaque cluster. Cette analyse nous permettra de positionner chaque cluster sur la matrice de Porter (PORTER et DE LAVERGNE [2003]). Elle offrira également de l'information aux animateurs.

Premier produit

La représentation en secteurs 5.3 a été réalisée à partir des effectifs de chaque cluster défini suite aux conclusions reprises sur le tableau 5.2.

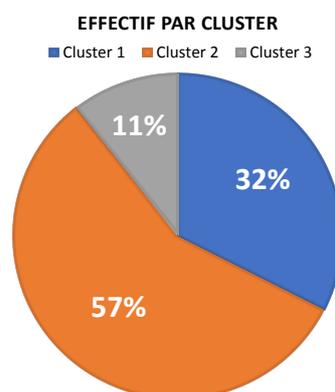


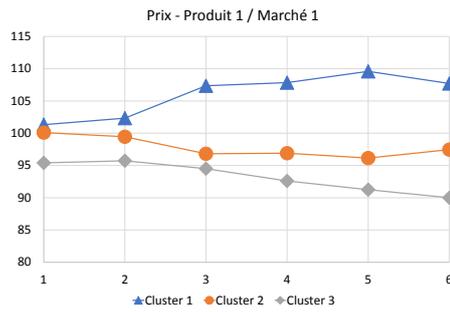
FIGURE 5.3 – Représentation des effectifs par cluster pour le produit 1

L'observation de la figure 5.3 nous montre que le cluster 2 est très représenté pour le produit 1 (P1). Il rassemble 57% des comportements. Ensuite, le cluster 1 rassemble 32% des joueurs et enfin le cluster 3 ferme la marche avec 11% des joueurs.

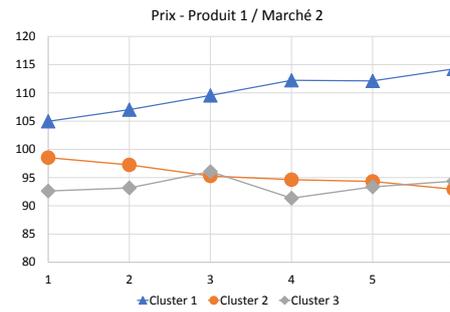
Ainsi, dans chaque partie, l'animateur a plus d'une équipe sur deux qui appartient au cluster 2.

Les figures 5.4a, 5.4b, 5.4c, 5.4d, 5.4e, 5.4f et 5.4g représentent au cours du temps l'évolution des variables de chaque produit pour chaque cluster et chaque marché. Ainsi, la figure 5.4a nous permet d'observer l'évolution de la variable *Prix* pour le produit 1 sur le marché 1 au cours des tours 1 à 6. Chaque série de données reprend le centroïde du cluster pour la variable illustrée. La valeur 100 est la valeur moyenne générale pour toutes les classes recalculée à chaque tour.

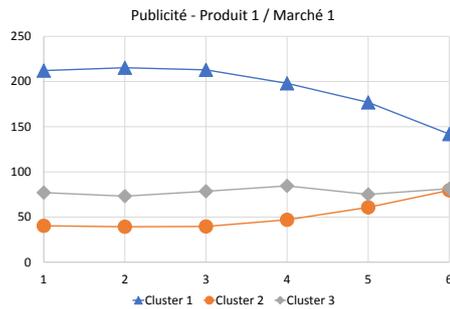
Pour rappel, le produit P1 est un produit de bas de gamme. Ainsi, la différenciation et la technologie n'ont aucun impact sur la décision du client. Cependant, une équipe qui investit en technologie profite, à terme, d'une réduction substantielle de ses coûts de production par une baisse du temps de fabrication unitaire, une rationalisation des consommations de matières premières et des coûts variables de production. Ce rappel nous permet d'établir un lien entre un positionnement prix du cluster 3 que nous considérerons comme agressif et un haut niveau technologique. En effet, ce cluster parvient



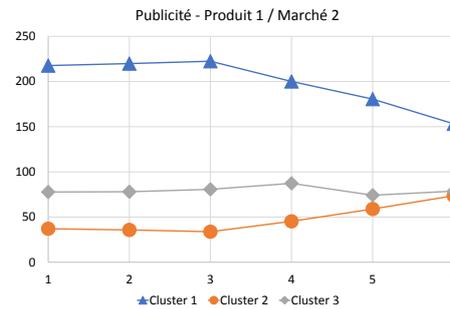
(a) Évolution de la variable **Prix** sur le marché 1



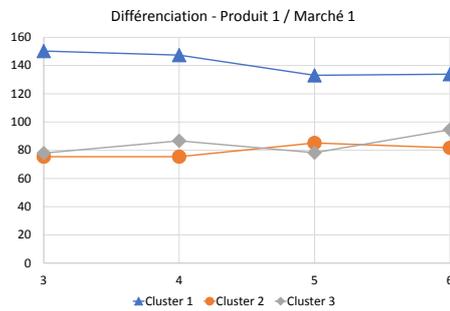
(b) Évolution de la variable **Prix** sur le marché 2



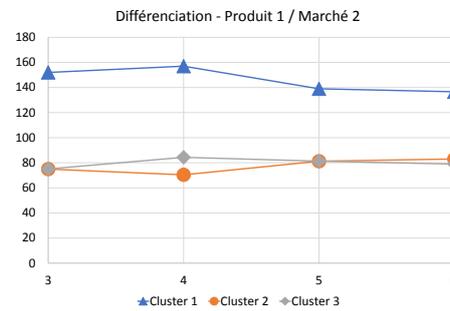
(c) Évolution de la variable **Publicité** sur le marché 1



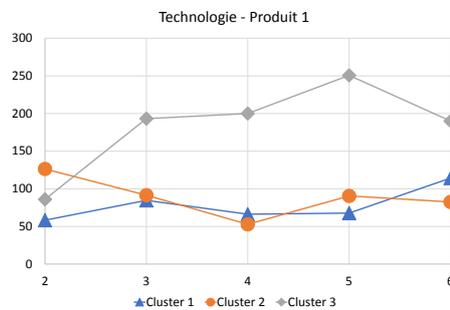
(d) Évolution de la variable **Publicité** sur le marché 2



(e) Évolution de la variable **Option** sur le marché 1



(f) Évolution de la variable **Option** sur le marché 2



(g) Évolution de la variable **Technologie**

FIGURE 5.4 – Représentation des variables décisionnelles associées au produit 1 par cluster sur Upstrat

à maintenir une marge commerciale malgré son prix bas grâce à la réduction des coûts opérée et à l'effet de volume qui découle de sa décision sur le critère *Prix*.

Le cluster 1 adopte un positionnement prix élevé qu'il compense par d'importants budgets publicitaires et une différenciation (fonctionnalités offertes aux clients) significative. Le cluster 2 réalise pour sa part moins d'efforts en technologie que le cluster 3. Cependant, pour les autres dimensions concurrentielles, il suit le cluster 3 sur tous les autres critères.

Nous résumons les observations réalisées qualitativement pour chaque cluster dans le tableau 5.3.

TABLEAU 5.3 – Synthèse des positionnements des joueurs par cluster pour le produit 1

	Cluster 1	Cluster 2	Cluster 3
Prix	élevé	médian	bas
Publicité	élevée	médiane basse	médiane
Différenciation	élevée	médiane	médiane
Technologie	médiane	médiane	très élevée

A la lumière de cette précision, nous pensons que les joueurs situés dans la classe 3 adoptent une stratégie de volume/coût au sens de Porter (PORTER et DE LAVERGNE [2003]). Leur représentation, de l'ordre de 11%, tend aussi à confirmer cette proposition. La classe 2, la plus représentée, regroupe tous les joueurs qui n'ont pas fait de choix clair sur ce produit. Nous pouvons les nommer les *suiveurs*. Enfin, la classe 1 regroupe les équipes souhaitant se différencier sur ce produit. Ce choix est assez discutable lorsque nous reprenons la typologie du produit.

Deuxième produit

Nous avons, pour le produit 2, réalisé les mêmes représentations graphiques et le même tableau de synthèse des comportements. Nous vous proposons de consulter dans un premier temps le graphique 5.5 afin de visualiser la répartition des effectifs par cluster pour le produit 2.

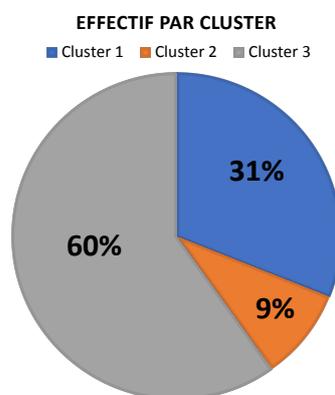


FIGURE 5.5 – Représentation des effectifs par cluster pour le produit 2

Le cluster 3 est représenté par 60% des joueurs. Le cluster 1 regroupe 31% des joueurs et enfin le cluster 2 rassemble 9% des joueurs restants. Les clusters 1 et 2 regroupent plus de 90% des joueurs sur une partie pour ce produit.

Les figures 5.6a, 5.6b, 5.6c, 5.6d, 5.6e, 5.6f et 5.6g reprennent sur les tours 2 à 6 les valeurs de chaque variable pour chaque centroïd de cluster. Ces représentations se lisent de la même manière que pour le produit 1.

Pour mémoire, nous rappelons quelques éléments du scénario de jeu joué. Le produit 2 est un produit de milieu de gamme. Il impose aux joueurs un bon positionnement prix et une présence commerciale significative. Une proposition technologique est également attendue par les clients. Il est assez difficile de se positionner durablement sur ce produit car il est très souvent au cœur de la stratégie des joueurs. De plus, il impose des investissements réguliers.

Le cluster 3 semble vouloir appliquer une stratégie de volume coût à ce produit en limitant les prix et en investissant peu sur la technologie et la différenciation. Le cluster 1 présente un caractère assez clairement orienté vers la différenciation au sens de Porter (PORTER et DE LAVERGNE [2003]). Les prix élevés sont compensés par des options de différenciation marquées. Elles offrent ainsi un bon rapport Qualité/Prix aux clients. Notons qu'un tiers des joueurs adoptent ce positionnement (figure 5.5). Les graphiques ne permettent pas de définir clairement le comportement des équipes affectées au cluster 2. Celui-ci ne représente que 9% des joueurs. Les positionnements sont médians sur tous les facteurs clés de succès hormis pour la technologie. Il semble que ces équipes aient changé d'orientation stratégique en cours de partie. Leur priorité a évolué et elles ont affecté leurs moyens à d'autres produits.

Le tableau 5.4 synthétise les observations réalisées.

TABLEAU 5.4 – Synthèse des positionnements des joueurs par cluster pour le produit 2

	Cluster 1	Cluster 2	Cluster 3
Prix	élevé	médian	bas
Publicité	élevée	médiane	basse
Différenciation	élevée	médiane	médiane
Technologie	médiane	élevée	très basse

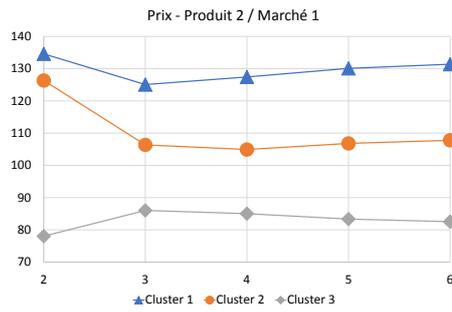
Ainsi, pour ce produit, 91% des comportements sont assimilables soit à une stratégie de volume/coût (Cluster 3), soit à un positionnement en différenciation (Cluster 1).

Troisième produit

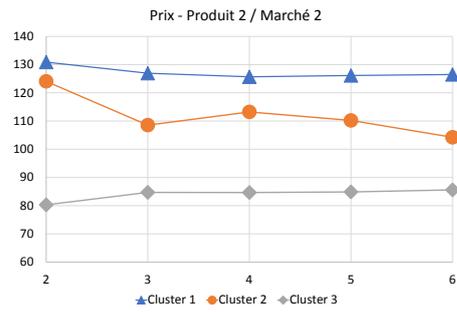
La répartition des joueurs par cluster est visible sur la figure 5.7. Le cluster 1 est majoritaire et représente 60% des joueurs. Le cluster 2 représente 30% des joueurs et enfin, le cluster 3 est très minoritaire avec uniquement 10% des joueurs.

Les comportements des trois clusters de joueurs pour le produit 3 sont illustrés par les graphiques 5.8a, 5.8b, 5.8c, 5.8d, 5.8e, 5.8f et 5.8g.

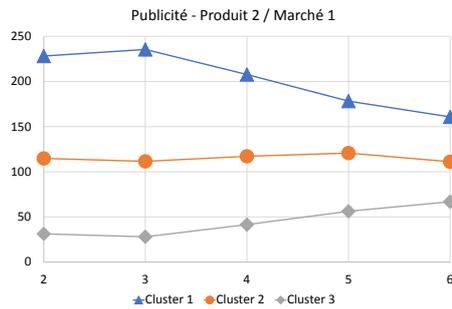
Afin de replacer ce produit dans le contexte économique, nous rappelons que le produit 3 est un produit *premium*, de très haut de gamme. Il n'intéresse qu'une fraction de la population, des passionnés souvent aisés. Il est aussi réservé aux clients précurseurs, à l'affût des dernières nouveautés et innovations. Ainsi, sur ce marché, il est inutile d'engager une guerre des prix. La technologie est prépondérante. Les options (différenciations) offertes sont aussi fondamentales pour mettre en place une vraie barrière d'entrée par rapport à la concurrence. Notons également que ce produit arrive tard sur le marché. Avec les données à notre disposition, il est difficile à caractériser. Il serait préférable d'attendre quelques tours de jeu complémentaires pour enrichir nos observations et ainsi améliorer cette analyse. Cependant, nous pouvons retrouver quelques caractéristiques majeures que nous vous proposons dans le paragraphe suivant.



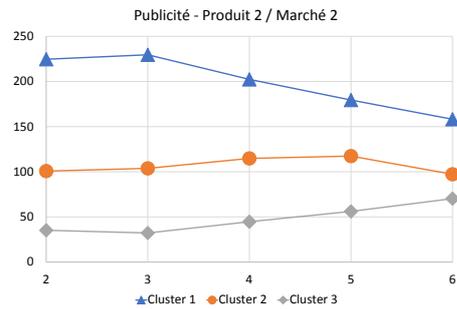
(a) Évolution de la variable **Prix** sur le marché 1



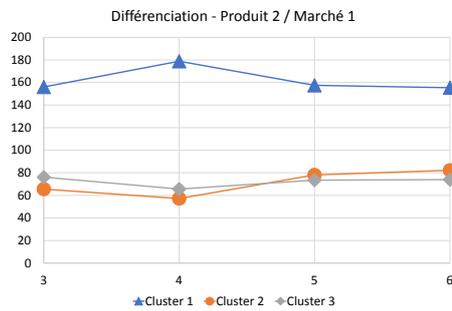
(b) Évolution de la variable **Prix** sur le marché 2



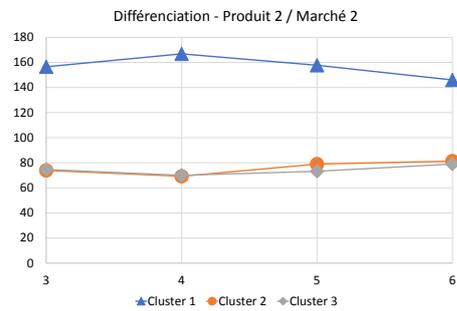
(c) Évolution de la variable **Publicité** sur le marché 1



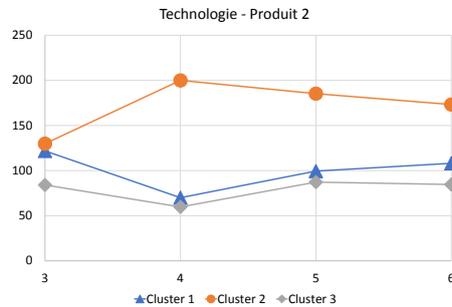
(d) Évolution de la variable **Publicité** sur le marché 2



(e) Évolution de la variable **Option** sur le marché 1



(f) Évolution de la variable **Option** sur le marché 2



(g) Évolution de la variable **Technologie**

FIGURE 5.6 – Représentation des variables décisionnelles associées au produit 2 par cluster sur Upstrat

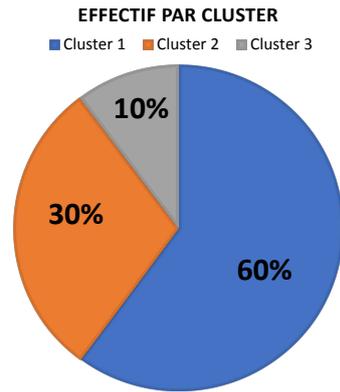


FIGURE 5.7 – Représentation des effectifs par cluster pour le produit 2

Le cluster 1 regroupe des équipes qui ne se sont pas lancées sur tous les marchés pour le produit 3 ou qui n’ont pas lancé dès sa sortie le produit. Ceci explique des valeurs faibles sur l’ensemble des critères du mix marketing. Le cluster 3 a, par contre, appliqué une vraie stratégie haut de gamme ou luxe. Il a fait les efforts nécessaires au leadership sur ce marché. Cela se manifeste par des prix de vente élevés, des efforts en publicité conséquents et une proposition technologique remarquable. Enfin, le cluster 2 est médian. Cela peut se définir par un « non-choix ». Il est souvent en retard par rapport aux meilleurs et ne prend pas les décisions à temps pour maintenir une position concurrentielle importante. Nous remarquons toutefois que les équipes souhaitant faire l’effort sur cette référence se démarquent nettement par rapport aux autres groupes.

Nous avons repris les orientations de chaque cluster dans le tableau 5.5.

TABLEAU 5.5 – Synthèse des positionnements des joueurs par cluster pour le produit 3

	Cluster 1	Cluster 2	Cluster 3
Prix	bas	médian élevé	élevé
Publicité	basse	élevée	élevée
Différenciation	basse	médiane	élevée
Technologie	médiane	médiane	élevée

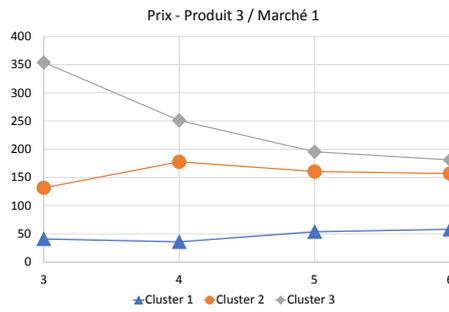
5.2.3 Classification

Dans le paragraphe précédent, nous avons souhaité donner un sens économique aux décisions des joueurs. Nous avons plus particulièrement observé les comportements moyens des individus de chaque cluster sur chaque produit. Ces observations nous ont permis de nous rapprocher des grandes stratégies reprises dans le modèle de Porter (PORTER et DE LAVERGNE [2003]).

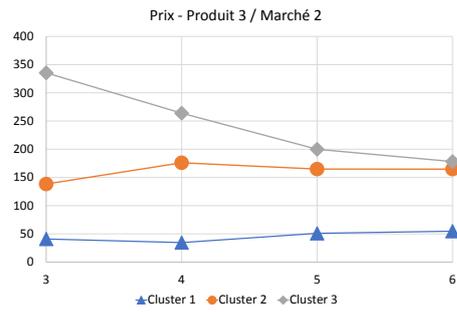
Nous considérons désormais que le partitionnement réalisé dans la section précédente constitue des classes de joueurs. Nous souhaitons maintenant classer une nouvelle équipe parmi les classes identifiées. Nous validerons les conclusions du chapitre 4 sur ce nouveau jeu de données.

Ainsi, dans cette partie, nous avons souhaité :

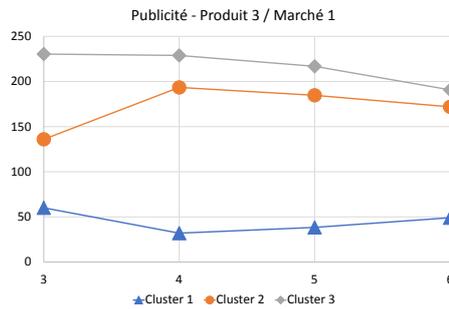
- identifier la méthode de classification la plus performante,
- valider qu’il était possible d’exploiter la classification des équipes avec des données de jeu partielles dès le début d’une partie.



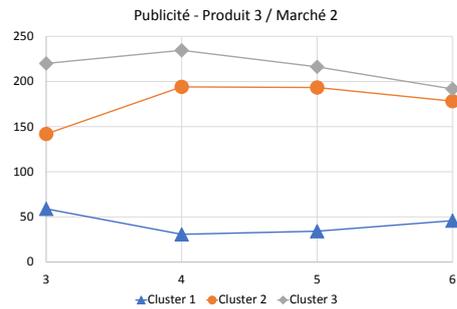
(a) Évolution de la variable **Prix** sur le marché 1



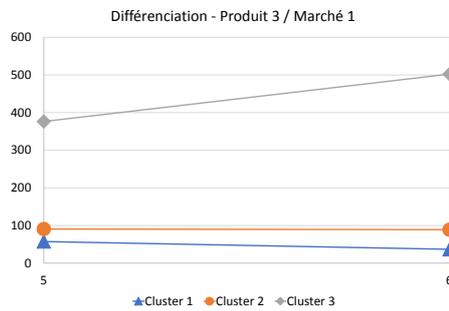
(b) Évolution de la variable **Prix** sur le marché 2



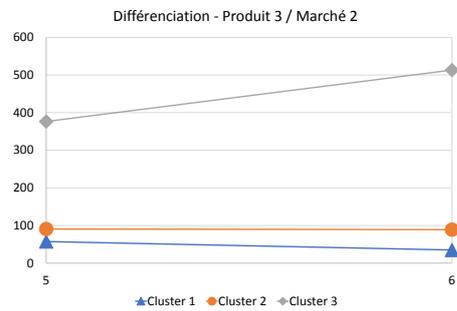
(c) Évolution de la variable **Publicité** sur le marché 1



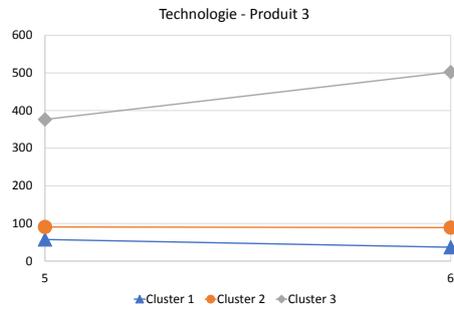
(d) Évolution de la variable **Publicité** sur le marché 2



(e) Évolution de la variable **Option** sur le marché 1



(f) Évolution de la variable **Option** sur le marché 2



(g) Évolution de la variable **Technologie**

FIGURE 5.8 – Représentation des variables décisionnelles associées au produit 3 par cluster sur Upstrat

Choix de la méthode de classification

Avec le même objectif que pour le clustering, nous souhaitons valider la qualité de la classification et vérifier si les meilleurs paramétrages trouvés pour FirStrat sont identiques pour Upstrat. Nous comparons ainsi différentes méthodes de classification. Pour chacune d'elles, nous avons, le cas échéant, proposé plusieurs paramétrages. Les essais de classification ont été réalisés sur les clusterings présentés dans le paragraphe précédent sur les fenêtres de décisions précitées (tableau 5.2).

Pour chaque méthode de classification, nous avons scindé le jeu de données en deux échantillons. Le premier échantillon, réservé à l'apprentissage, comprend les deux tiers des données. Le second échantillon a été utilisé pour le test de validation.

A l'issue de chaque classification sur le jeu de validation, nous avons comparé les classes données par ces tests aux classes issues du clustering. Nous avons ainsi obtenu un taux d'équipes bien classées pour chaque méthode de classification. Le tableau 5.6 reprend donc pour toutes les méthodes de classification, caractérisées par les lignes, les taux d'équipes bien classées de l'échantillon de test pour chaque produit, représentant les colonnes. En gras, nous avons mis en évidence les meilleurs taux d'équipes bien classées.

TABLEAU 5.6 – Taux de biens classés par méthode de classification

		P1	P2	P3
K-NN euclidienne	$k = 3$	92%	93%	96%
	$k = 5$	91%	92%	96%
	$k = 7$	91%	95%	91%
K-NN pondéré	$k = 3$	90%	93%	96%
	$k = 5$	92%	91%	97%
	$k = 7$	91%	92%	95%
Centroïd		83%	85%	89%
Régression logistique		76%	56%	85%

Le tableau 5.6 permet de confirmer que la méthode des K plus proches voisins (K-NN) offre de très bons résultats par rapport aux deux autres méthodes testées : la proximité au centroïd et la régression logistique. En effet, cette méthode propose un taux d'équipes bien classées supérieur à 90% indépendamment du paramétrage choisi. La méthode K-NN est donc conservée.

Aussi, nous souhaitons valider le paramétrage de cette dernière et avons évalué deux méthodes de calcul de distance. Nous avons également fait varier k entre 3 et 7. Les aspects théoriques de ces deux paramètres sont présentés dans le chapitre 4. Les résultats nous incitent à exploiter la distance pondérée. En effet, le taux d'équipes bien classées est en moyenne légèrement meilleur avec ce mode de calcul de distance. Nous proposons également de fixer k à 5.

Classification au moyen de données partielles

Le paragraphe précédent nous permet d'effectuer un choix éclairé sur la méthode de classification que nous devons conserver. Aussi, comme pour le jeu de données FirStrat, nous souhaitons valider sur les données UpStrat que nous pouvons exploiter le même clustering pour l'ensemble de l'exploitation quel que soit le tour de jeu à optimiser. Ainsi, nous avons réalisé de nouveaux essais de classification. Ces tests ont été faits par la méthode K-NN en distance pondérée et pour $k = 5$. Nous avons fourni en données d'entrées

des données partielles. Par exemple, nous avons réalisé une classification à partir des données de la fenêtre 2_3. Nous avons comparé les classes affectées aux classes données par le clustering sur les données 2_4. Cette comparaison nous permet donc de définir un taux d'équipes bien classées au moyen de données partielles pour la fenêtre de référence ici 2_4.

La classification a été réalisée sur l'ensemble du jeu de données (sans constitution d'échantillons). Le tableau 5.7 récapitule les résultats obtenus avec pour chaque ligne une nouvelle fenêtre de données et en dernière colonne le pourcentage d'équipes bien classées. Nous avons indiqué en gras les valeurs provenant de la validation de la méthode de classification du paragraphe précédent. Ce sont les seules valeurs réalisées avec un échantillon d'apprentissage et un autre de test.

TABLEAU 5.7 – Taux d'équipes bien classées avec les données partielles

Produit	Données utilisées par le clustering	Données utilisées pour la classification	Taux de bien classés
P1	2-4	2-2	73%
		2-3	87%
		2-4	92%
P2	2-4	2-2	71%
		2-3	89%
		2-4	91%
P3	4-6	4-4	78%
		4-5	84%
		4-6	97%

Le tableau 5.7 nous permet d'observer, comme attendu, que la qualité de la classification évolue de manière significative lorsque la quantité de données d'entrée fournies augmente et se rapproche de l'ensemble des données exploitées pour réaliser le clustering. Cependant, en début de simulation, avec peu de données entrantes, nous remarquons que la classification reste correcte. En effet, pour le produit P2, au minimum 71 % des équipes sont bien classées. Notons également que le produit P3, même avec une faible quantité de données offre un excellent taux d'équipes bien classées. Ce phénomène s'explique par la typologie du produit qui favorise un positionnement très marqué des équipes dès son lancement commercial sur le marché.

Ces expérimentations nous permettent de valider que le clustering défini dans le tableau 5.2 est suffisant pour classer correctement une nouvelle équipe dans la bonne classe en cours de jeu quel que soit le cycle à optimiser.

Pour les expérimentations suivantes, nous exploiterons toujours la classification par la méthode K-NN ($k = 5$) par la distance pondérée.

Dans le paragraphe suivant, nous nous axerons sur la construction des arbres d'anticipation pour les trois DAS identifiés (assimilés aux trois produits du simulateur). Ainsi, l'exploitation des arbres d'anticipation nous permettra de générer une décision « potentielle » de chaque concurrent.

5.2.4 Arbre d'anticipation

Pour rappel, l'exploitation de l'arbre d'anticipation se fait en cours de jeu, avant le processus d'optimisation. Les arbres d'anticipation sont construits en amont de l'optimisation pour une classe déterminée. Dans notre cas, ils sont différents pour chaque produit.

Ainsi, nous disposons donc de 3 produits \times 3 classes par produit soit 9 arbres d'anticipation à paramétrer au total.

Le paramétrage intervient pendant la construction des arbres. Ainsi, en cours de construction, ils sont générés en sélectionnant la variable présentant la plus petite variance ou au contraire, la variable présentant la plus grande variance. La variable sélectionnée définit le critère suivant lequel seront découpés les groupes d'individus sur chaque branche de l'arbre. Le critère de séparation définira qui, parmi les individus de la branche, sera dans le groupe gauche ou droit. Nous distinguons deux intervalles différents :

- option A : **Intervalle A1** = valeur supérieure à la médiane ou **Intervalle A2** = valeur inférieure à la médiane
- option B : **Intervalle B1** = valeur comprise autour de la moyenne encadrée par son écart-type ou **Intervalle B2** = valeur en dehors de l'intervalle 1

Distance du centroïd des clusters à la moyenne générale

L'objectif des arbres d'anticipation est de proposer des décisions pour un joueur en connaissant ses décisions passées. Nous souhaitons ici exploiter le clustering réalisé afin d'évaluer si la décision proposée et donc anticipée au moyen de la connaissance apportée par le clustering, nous permet de nous approcher de la décision réelle d'un joueur.

Comme dans le chapitre 4, les tableaux 5.8, 5.9 et 5.10 permettent de comparer deux paramétrages permettant de produire une décision prédite. Pour chaque paramétrage, nous avons calculé la distance entre la décision produite par ce paramétrage et la décision réelle du joueur. Nous obtenons ainsi deux valeurs (une pour chaque paramétrage). Nous avons ensuite comptabilisé le nombre d'occurrences pour lesquelles chaque paramétrage est plus proche de la décision du joueur réel. Tous les essais ont été menés sur les 350 équipes à notre disposition.

L'analyse du tableau 5.8 permet de comparer la décision prédite issue de la moyenne générale à la décision prédite issue du centroïd de chaque classe (après classification). Nous avons détaillé les résultats pour chaque tour de jeu. La lecture du tableau pour le tour 2 sur le produit 1 nous indique 67%. Cela signifie que la décision produite par le centroïd de la classe d'affectation est plus proche de la décision réelle du joueur dans 67% des cas que la décision produite par la moyenne générale. La première ligne du tableau reprend la moyenne générale sur l'ensemble des tours.

TABEAU 5.8 – Comparaison des paramètres : centroïd vs. moyenne générale

	Produit 1	Produit 2	Produit 3
Moyenne	68%	68%	72%
Tour 2	67%	68%	
Tour 3	68%	71%	63%
Tour 4	76%	74%	77%
Tour 5	66%	66%	73%
Tour 6	65%	62%	77%

Nous constatons qu'une décision prédite par le centroïd du cluster d'affectation est préférable à la décision issue de la moyenne générale. En effet, les taux présentés dans le tableau 5.8 sont toujours supérieurs à 50%.

Notons que pour le produit 3, nous ne disposons pas de données pour le tour 2. En effet, ce produit naît commercialement en tour 4. Cependant, les équipes ont la possibilité d'investir en technologie et publicité avant le lancement commercial par anticipation.

Nous pouvons conclure ce paragraphe en indiquant que le clustering apporte une meilleure décision projetée que la moyenne générale. Ce clustering est donc source de connaissances pour nos futurs déploiements.

Sélection du paramétrage de la construction de l'arbre d'anticipation pour chaque produit

Nous souhaitons dans ce paragraphe, identifier pour chaque produit le paramétrage qui offrira la plus grande proximité entre la décision prédite par l'arbre d'anticipation et la décision réelle du joueur.

Les tableaux 5.9 dressent les résultats obtenus après comparaison des paramétrages de construction des arbres. Nous avons identifié quatre paramétrages qui représentent chacun une colonne du tableau. Nous dressons la liste ci-dessous avec comme premier paramètre la méthode de sélection de variable suivie par l'opérateur de branchement :

- Variance minimale / Moyenne encadrée
- Variance minimale / Médiane
- Variance maximale / Moyenne encadrée
- Variance maximale / Médiane

Chaque décision prédite par l'arbre paramétré a été comparée à la décision réelle du joueur nous offrant ainsi une distance (d_1). Dans un second temps, nous avons calculé comme dans le paragraphe précédent, la distance entre la décision produite par le centroïd de la classe et la décision réelle du joueur pour obtenir une nouvelle distance (d_2). Nous avons ensuite comptabilisé le nombre d'occurrences où $d_1 \leq d_2$. Les résultats repris dans le tableau sont présentés en moyenne sur l'ensemble des tours de jeu et également par tour de jeu.

TABLEAU 5.9 – Comparaison des paramètres des arbres par rapport à la méthode des centroïds

Sélection de variable		$min(\sigma)$		$max(\sigma)$	
Opérateur de branchement		Moyenne	Médiane	Moyenne	Médiane
Produit 1	Moyenne	82%	66%	67%	64%
	Tour 3	76%	59%	61%	55%
	Tour 4	79%	63%	63%	59%
	Tour 5	85%	69%	72%	73%
	Tour 6	89%	72%	73%	70%
Produit 2	Moyenne	83%	68%	68%	64%
	Tour 3	74%	62%	61%	59%
	Tour 4	83%	65%	61%	59%
	Tour 5	88%	69%	76%	71%
	Tour 6	88%	75%	74%	66%
Produit 3	Moyenne	78%	61%	59%	55%
	Tour 5	73%	56%	57%	52%
	Tour 6	83%	67%	62%	57%

La première conclusion que nous pouvons formuler est que les arbres d'anticipation offrent une décision anticipée plus proche de la décision du joueur réel que la décision anticipée par le centroïd pour chacun des trois produits. En effet, dans 68% des cas, il est préférable de choisir la décision issue des arbres d'anticipation quel que soit le paramétrage de celui-ci.

Une autre conclusion que nous pouvons réaliser à l'issue de ces expérimentations est similaire à celle formulée dans le chapitre 4. En effet, il est préconisé de sélectionner la variable à la plus petite variance comme variable discriminante (les valeurs sont marquées en gras dans le tableau 5.9). La réalisation des sous-groupes à chaque étape de la construction des arbres d'anticipation sera pour sa part réalisée en choisissant la moyenne encadrée comme opérateur de branchement. Pour le produit 1, le taux passe de 66% pour la médiane à 82% pour la moyenne encadrée (pour la moyenne des observations). Pour le produit 2, ce taux passe de 68% à 83% toujours en moyenne des observations. Pour le produit 3, les observations sont du même ordre. Nous sélectionnerons pour la suite des essais ce paramétrage lors de la construction des arbres. Il sera identique pour tous les produits et l'ensemble des tours de jeu.

Enfin, nous pouvons noter une réelle amélioration des prédictions réalisées au cours de l'évolution des tours de jeu. En effet, pour chaque produit, la performance la plus faible est toujours la performance du premier tour prédit. Plus la partie avance, plus les performances de l'arbre prédictif offrent des résultats intéressants et donc meilleurs que le paramétrage « centroïd ». Cette observation tend à prouver que les équipes, une fois installées sur un marché, maintiennent leur cap et ne modifient qu'à la marge leurs décisions économiques. La qualité de la prédiction des arbres augmente aussi fortement avec la quantité de données historiques exploitées permettant d'aiguiller plus finement le processus lors du parcours de l'arbre.

Comparaison du paramétrage choisi à la moyenne générale pour chaque produit

Afin de valider le paramétrage sélectionné, nous avons souhaité comparer la décision générée par l'arbre d'anticipation paramétré comme indiqué précédemment à la décision anticipée par la moyenne générale du jeu de données.

Rappelons ici que nous exploiterons en début de partie (sur les deux premiers tours de jeu) la moyenne générale car l'arbre d'anticipation ne dispose pas de données d'entrées suffisantes pour être exploité. Il est d'ailleurs impossible, dès le début du jeu de classer les équipes dans les classes produites par le clustering.

Le tableau 5.10 présente une dernière comparaison entre le paramétrage sélectionné dans le paragraphe précédent et la décision anticipée par exploitation de la moyenne générale des données. Ces comparaisons ont été menées de la même façon que pour les deux paragraphes précédents

TABLEAU 5.10 – Comparaison des paramètres : Arbres d'anticipation paramétrés vs. Moyenne générale

	Produit 1	Produit 2	Produit 3
Tour 3	80%	79%	
Tour 4	83%	85%	
Tour 5	88%	90%	75%
Tour 6	93%	94%	87%

En début de jeu, dès le tour 3, il est, dans 8 cas sur 10, préférable d'exploiter les décisions anticipées par les arbres. 3 tours de jeu plus tard, cette statistique évolue favorablement en faveur de ces mêmes arbres pour atteindre plus de 9 cas sur 10.

Nous pouvons conclure que l'exploitation des arbres d'anticipation offre une décision approchée correcte dès lors qu'ils sont exploitables soit à partir du tour 3 pour les produits P1 et P2 et du tour 5 pour le produit P3.

5.3 Évaluation du processus global d'optimisation

Dans ce chapitre, nous avons dans un premier temps évalué la pertinence d'un clustering sur les données recueillies pour le simulateur UpStrat. Nous avons ensuite défini un clustering performant et lui avons associé une méthode de classification basée sur une méthode K-NN exploitant la distance pondérée que nous avons proposée. Nous avons également déployé avec succès les arbres d'anticipation qui ont été présentés dans le chapitre 4. Ceux-ci ont été paramétrés de façon idoine et ont permis, d'après les expérimentations établies, de montrer que leur utilisation semblait utile afin de mieux prédire la décision des concurrents en cours de processus.

Nous disposons donc, d'une part, d'un outil d'optimisation de décisions basé sur une hybridation de méthodes : PLNE, recherche locale. Nous disposons également d'un outil d'anticipation du comportement de concurrents permettant ainsi d'estimer, suivant les lois du marché, une demande prévisionnelle. La performance de cette méthode d'anticipation a été prouvée dans le paragraphe précédent.

Notre objectif est maintenant d'évaluer la performance globale de la méthode et de montrer l'apport de la méthode d'anticipation dans le processus d'optimisation. Ainsi, dans un premier temps, nous chercherons à évaluer la qualité de l'optimisation sans apport de connaissances. Dans un second ensemble d'expérimentations, nous effectuerons les mêmes essais en intégrant cette fois la connaissance via les arbres d'anticipation. Nous comparerons les approches et conclurons sur la qualité globale de la méthode proposée.

5.3.1 Optimisation sans apport de connaissances

Pour cette première partie dédiée aux tests finaux, nous testons la performance de la méthode d'optimisation hybridée sans apport de connaissances. Pour mener à bien cette étude, nous avons réalisé une sélection aléatoire de dix parties jouées dans des conditions identiques. Ainsi, toutes les parties ont exploité le même scénario. Huit équipes étaient engagées a minima. Les simulations se sont déroulées sur sept tours de jeu.

Afin de pouvoir calculer les demandes prévisionnelles nécessaires à l'optimisation, nous utilisons les données moyennes de l'ensemble de la base de connaissances.

Nous avons souhaité, pour chaque jeu de nos données, jouer successivement à la place des cinq premières équipes de chaque tour pour les tours 1 à 6. Ainsi, nous nous sommes positionnés au tour 1 et avons remplacé la première, puis la deuxième (la première étant retournée dans le jeu) jusqu'à la cinquième de l'année 1. Ensuite, nous avons effectué le même remplacement pour le tour 2 et ainsi de suite jusqu'au tour 6. Signalons qu'il est probable que la première équipe du tour 1 soit différente de la première équipe du tour 2. Nous n'avons pas suivi les performances des équipes. Notre méthode de remplacement nous permet de nous affranchir de la vie d'une équipe en particulier. Nous préférons nous confronter à la meilleure équipe, à la seconde équipe... Pour chaque remplacement, dix exécutions ont été faites. Nous dénombrons donc 600 exécutions par position au classement. Au total, 3 000 exécutions ont été nécessaires pour pouvoir valider nos résultats.

Lors de chaque exécution, nous relevons le rang obtenu par notre approche. Ainsi, il nous est possible d'évaluer si le processus d'optimisation est parvenu à améliorer la performance de l'équipe réelle qu'il a remplacé, égaler sa performance ou au contraire de détériorer son classement. Pour rappel, chaque remplacement a été réalisé 10 fois de suite dans les mêmes conditions économiques et concurrentielles.

La figure 5.9 présente le boxplot des classements obtenus par l'agent d'optimisation

après remplacement de la première équipe sur les six premiers tours de jeu.

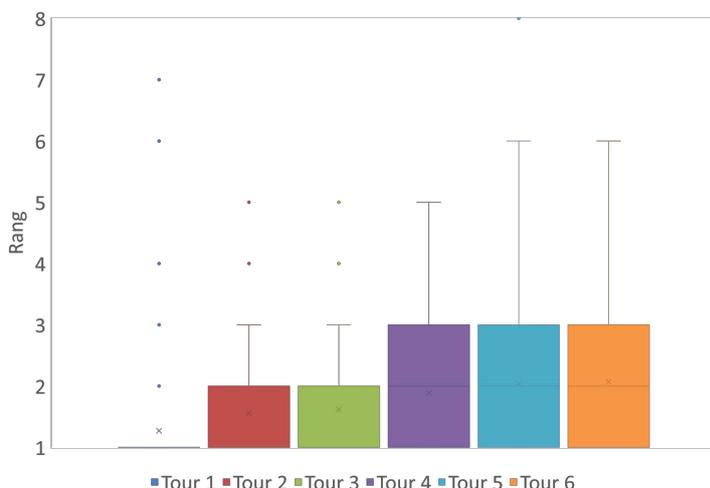


FIGURE 5.9 – Rang obtenu après remplacement de la première équipe (sans apport de connaissances)

Nous visualisons, sur la figure 5.9, que l’optimisation parvient à égaler son classement sur le premier tour dans de nombreux cas. Nous observons également que plus les tours de jeu avancent, plus les alternatives offertes aux joueurs s’enrichissent et plus l’agent rencontre des difficultés pour réitérer ses excellentes performances du premier tour. Sur les tours 2 et 3, l’agent se positionne régulièrement entre la première et la deuxième place. En fin de partie, il occupe toujours dans la plupart des cas le podium.

Le tableau 5.11 présente des taux assimilables à une performance. Pour tous les tableaux du même type (5.11, 5.12, 5.13, 5.14 et 5.15) nous avons souhaité quantifier la capacité du processus d’optimisation à améliorer, conserver ou au contraire dégrader la performance d’une équipe réelle. Par exemple, sur le tableau 5.11, pour le tour 1, dans 85% des exécutions, le processus d’optimisation a une performance identique à l’équipe remplacée. Dans ce cas, le processus est parvenu à se hisser à la première place du classement. Sur la ligne en dessous, nous observons que dans 15% des exécutions, le processus s’est classé après la première place. La performance est donc dégradée par rapport à la performance de l’équipe remplacée.

TABLEAU 5.11 – Répartition des rangs obtenus après remplacement de la première équipe (sans apport de connaissance)

	Tour 1	Tour 2	Tour 3	Tour 4	Tour 5	Tour 6
Rang identique	85%	64%	65%	40%	45%	42%
Rang dégradé	15%	36%	35%	60%	55%	58%

Le tableau 5.11 nous indique qu’en fin de partie, l’agent, bien que sans capacité d’anticipation, arrive en première position dans 42% des cas. Les dégradations des résultats semblent naturelles car il est impossible de nous positionner avant le premier.

La figure 5.10, quant à elle, nous permet d’apprécier le boxplot des rangs obtenus après remplacement de la seconde équipe. Nous avons marqué par une ligne noire horizontale le classement obtenu par l’équipe de référence (l’équipe qui a été remplacée). Pour les quatre premiers tours, la conclusion est similaire à la conclusion précédente.

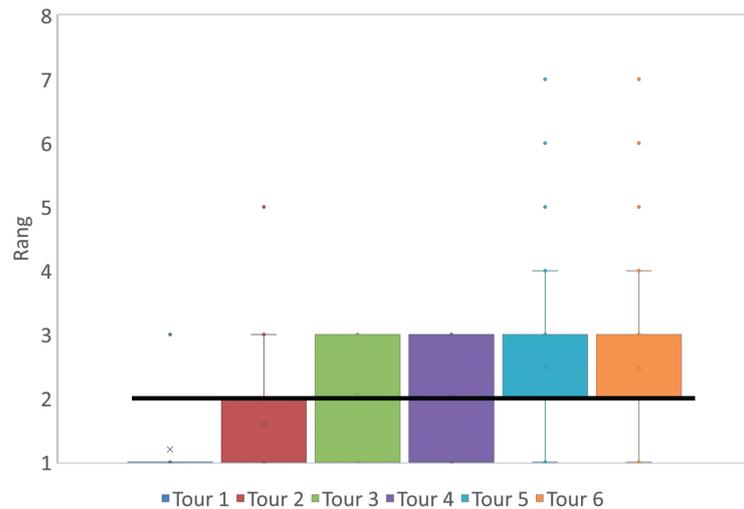


FIGURE 5.10 – Rang obtenu après remplacement de la deuxième équipe (sans apport de connaissances)

Nous observons qu’à partir du tour 5, il est plus rare que l’algorithme se hisse en première position. Notons également que l’agent est capable d’améliorer le rang obtenu par le joueur réel.

TABEAU 5.12 – Répartition des rangs obtenus après remplacement de la seconde équipe (sans apport de connaissances)

	Tour 1	Tour 2	Tour 3	Tour 4	Tour 5	Tour 6
Rang amélioré	81%	56%	28%	31%	23%	22%
Rang identique	18%	30%	40%	37%	28%	30%
Rang dégradé	1%	14%	32%	32%	49%	48%

Le tableau 5.12 nous indique que l’agent atteint en tour 1 la première marche du podium dans 81% des cas. Il améliore donc le rang de l’équipe qu’il a remplacée. On observe qu’il est de plus en plus difficile d’améliorer la performance au fur et à mesure de l’avancée des tours de jeu. En fin de simulation, l’agent se situe derrière l’équipe remplacée dans 1 cas sur 2.

La figure 5.11 a été réalisée suite au remplacement de l’équipe qui a atteint la troisième place. Ici, nous remarquons que nos processus d’optimisation parviennent dans la plupart des cas à égaler voire améliorer la performance des joueurs réels. Cette conclusion est vraie pour les premiers tours. Les difficultés augmentent avec l’avancée des tours de jeu.

TABEAU 5.13 – Répartition des rangs obtenus après remplacement de la troisième équipe (sans apport de connaissances)

	Tour 1	Tour 2	Tour 3	Tour 4	Tour 5	Tour 6
Rang amélioré	97%	83%	61%	59%	45%	49%
Rang identique	2%	15%	36%	34%	31%	30%
Rang dégradé	1%	2%	3%	7%	24%	21%

Le tableau 5.13 confirme la figure. Cependant, aux tours 5 et 6, nous observons que la performance de l’agent reste très honorable. En effet, dans 76% et 79% des cas, il égale à minima la performance du joueur réel.

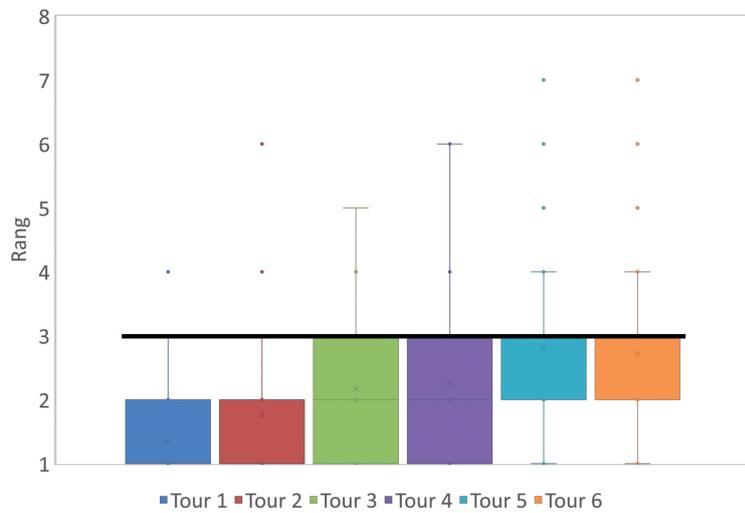
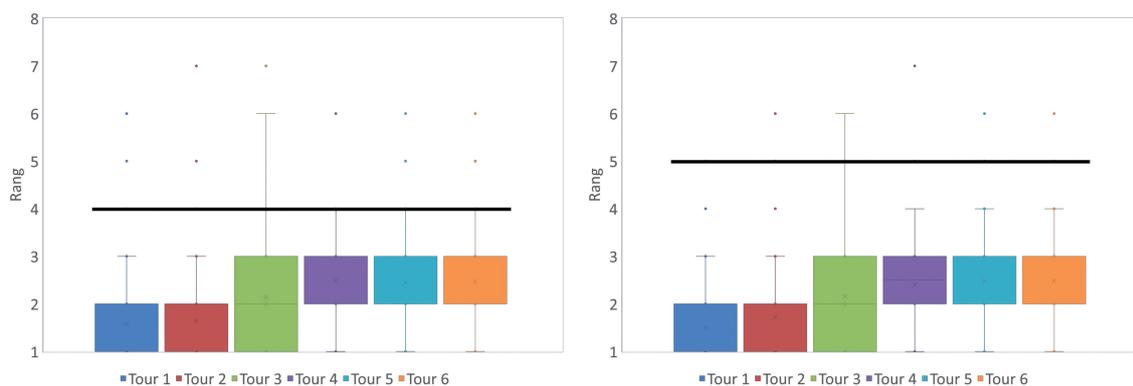


FIGURE 5.11 – Rang obtenu après remplacement de la troisième équipe (sans apport de connaissances)

FIGURE 5.12 – Rang obtenu après remplacement de la quatrième et de la cinquième équipe (sans apport de connaissances)



(a) Rang obtenu après remplacement de la quatrième équipe (sans apport de connaissances)

(b) Rang obtenu après remplacement de la cinquième équipe (sans apport de connaissances)

Les figures 5.12a et 5.12b et les tableaux 5.14 et 5.15 confirment la tendance précédente. En effet, le processus d'optimisation atteint un meilleur classement que l'équipe remplacée dans plus de 82% des cas si l'on remplace la quatrième équipe. Lorsque l'on remplace la cinquième équipe, le processus d'optimisation se classe avant l'équipe qu'il a remplacée dans plus de 90% des exécutions.

TABLEAU 5.14 – Répartition des rangs obtenus après remplacement de la quatrième équipe (sans apport de connaissances)

	Tour 1	Tour 2	Tour 3	Tour 4	Tour 5	Tour 6
Rang amélioré	96%	96%	97%	97%	85%	82%
Rang identique	2%	2%	1%	2%	9%	11%
Rang dégradé	2%	2%	2%	1%	6%	7%

TABLEAU 5.15 – Répartition des rangs obtenus après remplacement de la cinquième équipe (sans apport de connaissances)

	Tour 1	Tour 2	Tour 3	Tour 4	Tour 5	Tour 6
Rang amélioré	97%	97%	97%	96%	91%	91%
Rang identique	3%	1%	2%	3%	4%	4%
Rang dégradé	0%	2%	1%	1%	5%	5%

Nous pouvons formuler une autre conclusion. Le processus d'optimisation sans apport de connaissances présente de bonnes performances au regard des équipes remplacées. Cependant, les figures nous indiquent également que notre processus, lorsqu'il remplace des équipes plus éloignées du podium, peine à jouer les premiers rôles sur le marché concurrentiel. Cela s'explique par le fait qu'il est en concurrence avec d'autres équipes performantes.

5.3.2 Optimisation avec apport de connaissances

Les expérimentations menées dans le paragraphe précédent ont permis de mettre en évidence la performance de l'agent décisionnel au cours du jeu. Cette performance, bien que parfois perfectible, s'est avérée dans la plupart des cas aussi bonne voire meilleure que les résultats obtenus par les équipes réelles. Nous avons aussi remarqué que l'avancée dans la partie avait tendance à réduire la performance de la méthode et la rendait de temps en temps perfectible. Cette observation est, de notre avis, occasionnée par l'augmentation du nombre de choix stratégiques disponibles aux joueurs.

Aussi, nous pensons qu'il est possible d'améliorer cette performance en intégrant les décisions proposées par les arbres d'anticipation présentés dans le chapitre 4. Nous souhaitons vérifier l'hypothèse suivante : « La meilleure anticipation des décisions concurrentielles permet une meilleure décision économique de l'agent ».

L'exploitation des décisions anticipées vient donc remplacer les données moyennes issues de l'historique archivées dans la base de connaissance. Ainsi, avant chaque optimisation, nous produisons les décisions pour chaque équipe en nous basant sur les décisions antérieures de ce même individu. Ces décisions proviennent des arbres qui sont eux-mêmes spécifiques à la classe d'affectation de l'équipe à simuler. Ainsi, nous devons disposer de l'historique et donc du chemin emprunté par l'équipe afin de la classer et d'en anticiper le comportement.

Nous avons donc réalisé les mêmes expérimentations que précédemment. Cependant, nous avons débuté le remplacement à partir du tour 3. La prédiction au tour 2 n'offre que les décisions du tour 1 comme information pour le parcours de l'arbre. Ce manque d'information assimile la décision prédite à la moyenne du centroïd.

Les figures ci-dessous présentent les boxplots obtenus lors de ces expérimentations au cours desquelles, nous avons de nouveau remplacé successivement différentes équipes.

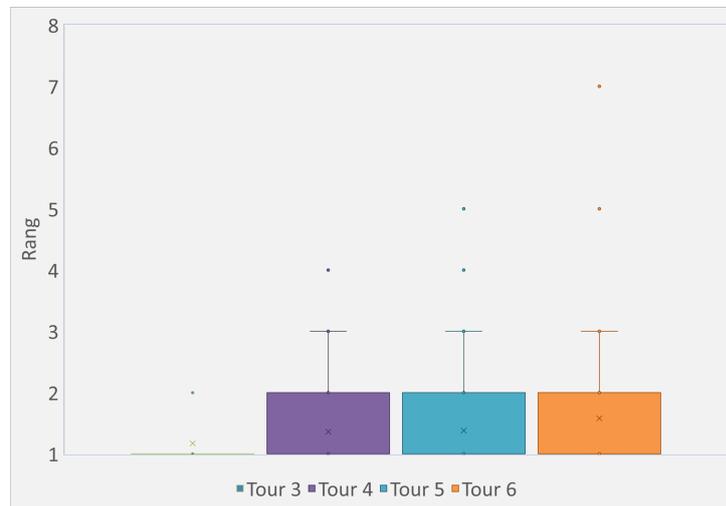


FIGURE 5.13 – Rang obtenu après remplacement de la première équipe (avec apport de connaissances)

La figure 5.13 et le tableau 5.16 présentent les excellentes performances du processus pour le tour 3. En effet, l'agent parvient à égaler la performance du joueur dans 83% des situations. La dégradation de la performance est peu visible pour les tours 4, 5 et 6.

TABLEAU 5.16 – Répartition des rangs obtenus après remplacement de la première équipe (avec apport de connaissances)

	Tour 3	Tour 4	Tour 5	Tour 6
Rang identique	83%	70%	73%	62%
Rang dégradé	17%	30%	27%	38%

Le tableau 5.16 nous indique que nous égalons la performance de l'équipe remplacée lorsque nous intégrons de la connaissance. Même en remplaçant le premier, nous sommes encore en mesure au tour 6 d'égaliser la performance du premier dans 62% des cas. Sans apport de connaissance, ce taux n'atteignait que 42% (cf. tableau 5.11).

Le remplacement du joueur classé en deuxième position est visible sur la figure 5.14. Ici, les performances sont aussi très bonnes et le processus d'optimisation se classe dans plus de 90% dans une position identique ou meilleure au joueur réel pour les tours 3 à 5 (tableau 5.17). En tour 6, cette performance se réduit à 86% mais reste excellente.

Lorsque nous avons remplacé les joueurs ayant atteint la troisième place du classement pour les tours 3 à 6 (figure 5.15), nous avons remarqué une réelle amélioration du classement offert par l'agent décisionnel. En effet, ce dernier améliore ou égale la position du joueur à hauteur de 88% pour le tour 3, 89%, 82% et 79% respectivement pour les tours 4 à 6 (tableau 5.18).

Sur les deux dernières figures 5.16a et 5.16b, nous observons que l'agent met à distance les équipes réelles. Les classements dégradés sont marginaux. Dans plus de 95%

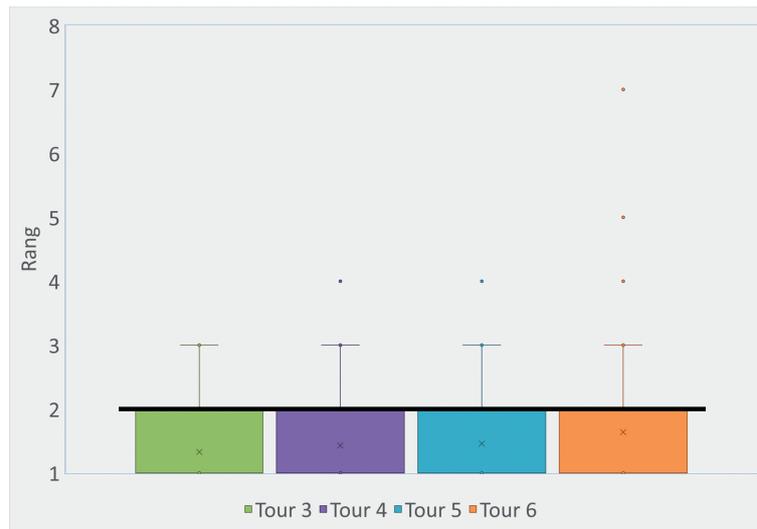


FIGURE 5.14 – Rang obtenu après remplacement de la deuxième équipe (avec apport de connaissances)

TABLEAU 5.17 – Répartition des rangs obtenus après remplacement de la deuxième équipe (avec apport de connaissances)

	Tour 3	Tour 4	Tour 5	Tour 6
Rang amélioré	74%	66%	68%	58%
Rang identique	19%	28%	22%	28%
Rang dégradé	7%	6%	10%	14%

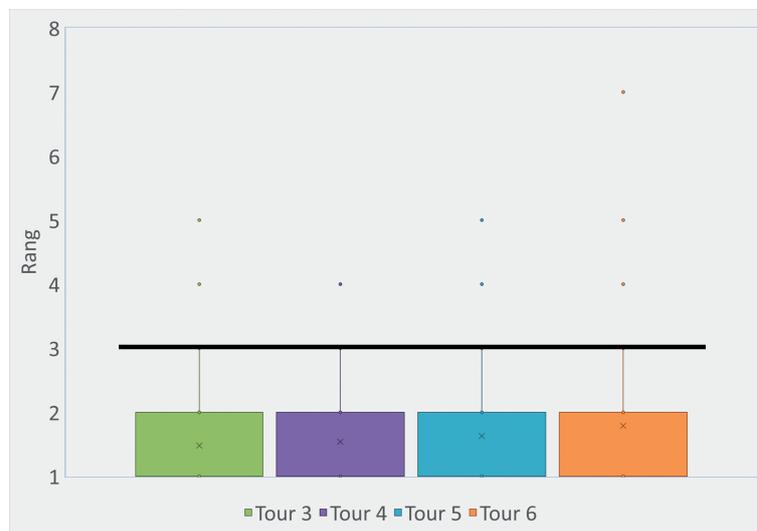
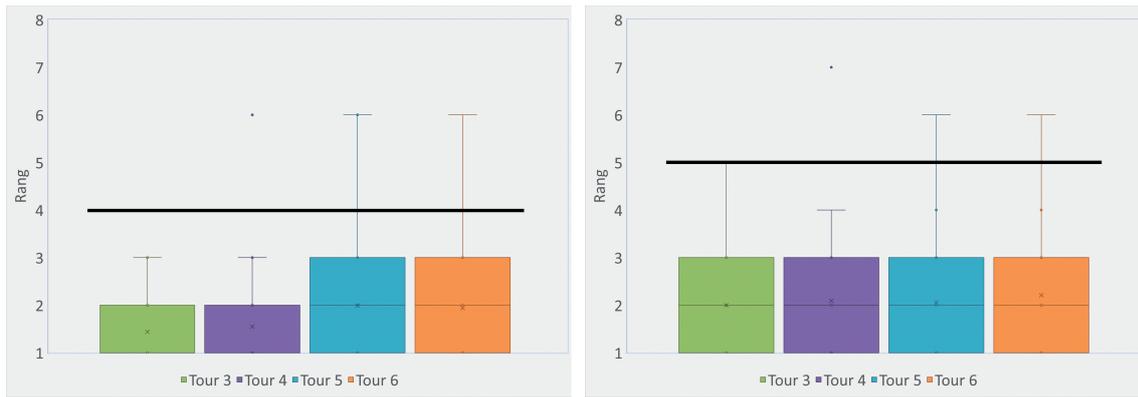


FIGURE 5.15 – Rang obtenu après remplacement de la troisième équipe (avec apport de connaissances)

TABLEAU 5.18 – Répartition des rangs obtenus après remplacement du troisième (avec apport de connaissances)

	Tour 3	Tour 4	Tour 5	Tour 6
Rang amélioré	88%	89%	82%	79%
Rang identique	10%	6%	11%	15%
Rang dégradé	2%	5%	7%	6%



(a) Rang obtenu après remplacement de la quatrième équipe (avec apport de connaissances)
 (b) Rang obtenu après remplacement de la cinquième équipe (avec apport de connaissances)

des cas, les agents prennent de meilleures décisions économiques que les participants réels (tableaux 5.19 et 5.20).

TABLEAU 5.19 – Répartition des rangs obtenus après remplacement de la quatrième équipe (avec apport de connaissances)

	Tour 3	Tour 4	Tour 5	Tour 6
Rang amélioré	100%	99%	97%	92%
Rang identique	0%	0%	1%	7%
Rang dégradé	0%	1%	2%	1%

TABLEAU 5.20 – Répartition des rangs obtenus après remplacement de la cinquième équipe (avec apport de connaissances)

	Tour 3	Tour 4	Tour 5	Tour 6
Rang amélioré	98%	99%	98%	95%
Rang identique	2%	0%	1%	4%
Rang dégradé	0%	1%	1%	1%

5.4 Conclusions sur les expérimentations

Dans les paragraphes précédents, nous avons souhaité évaluer dans un premier temps, la performance du processus d'optimisation dans son ensemble sans apport de connaissances. Cette étape était indispensable. Elle nous a permis de valider la bonne tenue des résultats du processus complet. Ces conclusions sont d'autant plus intéressantes que dans une première série d'expérimentations, les agents prenaient des décisions sans traitement spécifique apportant de la connaissance.

Dans un second temps, nous avons souhaité exploiter les données des arbres d'anticipation afin d'améliorer l'estimation des ventes prévisionnelles des agents. Nous espérons ainsi faire progresser les agents par apport de connaissances.

Les différentes expérimentations et paramétrages nous permettent donc de valider deux éléments fondamentaux. Premièrement, il est possible de faire appel à un processus

d'optimisation dans un contexte économique et concurrentiel afin de prendre des décisions éclairées dans le cadre d'un business game. Deuxièmement, l'apport de connaissances dans le processus permet d'améliorer considérablement les performances de la méthode proposée. Nous pouvons également ajouter qu'un agent autonome nourri par une connaissance performante et structurée (au moyen des arbres par exemple) est capable de devancer dans de très nombreux cas des joueurs réels.

Aussi, nous avons indiqué dans les différentes expérimentations menées qu'il n'était pas toujours possible de déployer les arbres d'anticipation. En effet, ces arbres demandent des données en entrées, correspondant à l'historique des équipes, pour parcourir les branches et ainsi aboutir à une proposition de décisions. En début de simulation, cette condition n'est pas remplie. Nous proposons donc d'apporter de la connaissance au moyen de deux autres méthodes présentées précédemment :

- l'exploitation de la **moyenne générale**,
- l'exploitation des **centroïds**.

Dans le paragraphe dédié au paramétrage des arbres d'anticipation, nous avons prouvé que l'exploitation des centroïds des classes à des fins d'anticipation de décisions concurrentielles offrait une meilleure réponse que l'exploitation de la moyenne générale des données Upstrat. Cependant, pour permettre l'utilisation des centroïds, il est indispensable de connaître la classe de l'équipe. Or, la classification impose une connaissance des décisions historiques de l'équipe. Ainsi, l'exploitation du centroïd ne pourra se faire qu'à partir du tour 2. Avant cette période de jeu, seules les données stockées sont exploitables.

Nous proposons donc une stratégie générale d'exploitation des outils permettant d'anticiper les décisions des concurrents en cours de partie que nous récapitulons dans le tableau 5.21.

TABLEAU 5.21 – Préconisation de l'outil d'anticipation de décisions concurrentielles

	Tour 1	Tour 2	Tour 3	Tour 4	Tour 5	Tour 6
Prédiction par ...	Moyenne	Centroïd	Arbre	Arbre	Arbre	Arbre

Conclusion et perspectives

Dans ce chapitre, nous rappellerons dans un premier temps les contributions principales de cette thèse en les situant par rapport aux objectifs annoncés dans l'introduction et à ceux de Strat&Logic. Dans un second temps, nous présenterons les perspectives que nous souhaitons décliner sur deux aspects. Nous identifierons ainsi des propositions de prolongement de la méthode. Nous nous interrogerons également sur l'exploitation pédagogique de ces travaux de recherche.

Contributions

La première partie de cette thèse s'intéresse à la modélisation de l'entreprise comme représentée dans le business game UpStrat. Elle s'intéresse également à la construction d'ensembles de décisions par méthode d'optimisation. A cet effet, les contributions de cette partie sont :

- Une analyse des **fonctions opérationnelles** principales de l'entreprise simulée par UpStrat. Chaque fonction a été identifiée et les liens entre elles ont été mis en lumière afin d'en autoriser la modélisation. L'opportunité d'une décomposition des fonctions et produits de l'entreprise a aussi été discutée. Deux procédures de choix, l'une par une approche stratégique et l'autre par un travail en comptabilité de gestion, ont été présentées.
- Une **modélisation des départements Production, Distribution et Finance**. Leurs optimisations par la mise en place de programmes linéaires en nombres entiers (PLNE) ont permis de proposer un ensemble de décisions opérationnelles optimales. Nous avons validé ces modèles en offrant aux processus d'optimisation les données d'entrée issues d'un contexte économique connu.
- Un **mode de sélection de solutions** en lieu et place de la fonction objectif. Notre problème d'optimisation combinatoire n'avait pas de critère d'optimisation évident pouvant être décliné en fonction objectif. Cependant, puisque la comparaison entre solutions est possible, dans un contexte donné, nous avons proposé de sélectionner les solutions à l'aide d'un tournoi. Ainsi, en nous basant sur la simulation UpStrat, nous avons proposé les règles de ce tournoi. Celui-ci a permis de retenir les meilleures solutions réalisables de l'espace de recherche tout en assurant le respect aux règles du jeu.
- Une **méthode de recherche locale**. Cette heuristique a permis de réaliser un parcours efficace de l'espace de recherche préalablement défini. Une fois correctement paramétrée, elle a offert de très bonnes performances en respectant la contrainte de temps d'exécution.

La deuxième partie de cette thèse s'intéresse à l'extraction de connaissances à partir d'historiques de jeu, à sa formalisation et son intégration en cours d'optimisation. A

cet effet nous avons proposé une méthodologie permettant d'anticiper le jeu des concurrents. Elle était nécessaire pour pouvoir estimer les ventes prévisionnelles de la solution en cours de construction. Pour mettre en œuvre cette méthodologie, les différentes contributions sont :

- La réalisation d'**un clustering performant**. Nous avons préparé nos données et avons déployé deux méthodes de clustering différentes. Nous avons proposé une mesure de la qualité de ce dernier et avons opté pour la méthode paramétrée la plus qualitative.
- Une **classification** effectuée au moyen d'un calcul de distance pondérée. En effet, afin de rendre plus impactantes les dernières décisions des équipes, nous avons souhaité mettre en œuvre un modèle de calcul de distance adapté à notre contexte d'étude. Après validation, celui-ci s'est avéré pertinent et a permis de classer correctement une nouvelle équipe dans la bonne classe.
- La **proposition de construction d'arbres d'anticipation** permettant une meilleure exploitation de la connaissance traitée par le clustering. En effet, nous avons proposé un processus de création d'arbre, qui, à partir des données historiques d'une équipe, permet d'anticiper la décision future. Le paramétrage a donné lieu à un terrain d'expérimentations. La performance de ces derniers a été prouvée en comparant la décision prédite d'une équipe au moyen d'un arbre d'anticipation à la décision réelle de cette même équipe.
- Une **intégration de connaissances** au cours du processus d'optimisation. Nous avons déployé le processus d'optimisation avec et sans apport de connaissances. Nous en avons évalué l'intérêt et avons proposé en fin de document un usage optimal de l'intégration de telles connaissances pendant le processus d'optimisation.

Perspectives

Vers une généralisation de la méthode ?

Les résultats présentés dans le chapitre 5 nous incitent à envisager une éventuelle généralisation de la méthode aux autres simulateurs du marché. En effet, nous sommes parvenus à prouver, sur un premier simulateur FirStrat, que les arbres d'anticipation permettaient de proposer des décisions économiques cohérentes ou au moins proches de la logique des utilisateurs si l'on connaît leur histoire. Cette conclusion a également été vérifiée sur le simulateur UpStrat en ayant préalablement choisi de scinder l'activité en Domaines d'Activités Stratégiques. Le processus d'optimisation couplé à une recherche locale ont également permis, dans ce contexte, d'identifier des décisions économiquement performantes sur les environnements économiques proposés.

Ainsi, nous pensons raisonnablement qu'il est envisageable d'appliquer la démarche à d'autres simulateurs. Cette démarche sera scindée en trois grandes étapes :

1. Une optimisation exacte de certaines parties du problème
2. Une méthode d'optimisation de type métaheuristique
3. Une anticipation du marché par extraction de connaissances

Les méthodes disponibles pour réaliser une optimisation exacte sont multiples et variées. La bibliographie en recherche opérationnelle est riche et offre bon nombre d'outils transposables aux modélisations d'activités économiques. Aussi, notre approche a donné de bons résultats. Il semble donc logique de proposer une approche généralisée similaire

au modèle déployé. Une hybridation des méthodes de recherche composée de méthodes exactes, la PLNE dans notre cas, et de métaheuristique, la recherche locale pour nous, est pertinente. Ce type de méthode à base de voisinage est très bien adaptée car seule une comparaison entre différentes solutions est possible.

Pour l'anticipation du marché, nous estimons que les arbres d'anticipation offrent de belles dispositions et de belles perspectives. Ils nous permettent de nous approcher ou de « coller » aux réactions des équipes. Cependant, la phase préliminaire de recueil des données et de clustering revêt un aspect fondamental.

Nous pensons aussi qu'il serait opportun d'intégrer le plus rapidement possible la notion de **décomposition**. Cette division ou séparation a été abordée lors des phases de modélisation. Nous avons découpé l'entreprise en départements s'approchant des fonctions opérationnelles classiques des entreprises. Dans la phase d'anticipation du marché, nous inspirant des méthodes d'analyse stratégique, nous nous sommes également orientés vers une décomposition de l'entreprise en secteurs concurrentiels distincts.

La décomposition d'un problème complet et complexe en sous-problèmes présente ainsi plusieurs avantages :

- *une facilité de résolution* : c'est évidemment le premier et principal atout de la méthode. Cette scission peut aussi réduire de façon importante la complexité et donc le temps d'exécution des processus. En effet, la réduction du nombre de variables entraîne une réduction des combinaisons possibles et ainsi évite l'explosion combinatoire.
- *un recours aux différentes sous-solutions à la demande* : chaque sous-problème admet une solution ou un ensemble de solutions. Ces solutions sont ensuite transférées au sous-problème suivant ou à un centre de régulation. Aussi, il est possible d'extraire uniquement les solutions d'un sous-problème. Cette opportunité prend un sens dans une configuration pédagogique. La mise à disposition de résultats partiels permet d'orienter les participants dans la bonne direction sans dévoiler le résultat final.

Le principal inconvénient de ce choix réside dans le caractère non-optimal de la solution et surtout des biais intégrés afin de permettre la décomposition. Cependant, dans les business games, une bonne solution est suffisante. Une solution optimale n'a pas réellement de sens puisqu'elle dépend du contexte économique et des concurrents qui évoluent sans cesse et simultanément.

Perspectives pédagogiques

L'usage de l'optimisation

Les perspectives pédagogiques sont nombreuses pour ce type d'outils. Nous proposons de séparer les utilisateurs en deux catégories. Nous trouverons d'une part les animateurs et enseignants et d'autre part les participants et stagiaires.

Pour un animateur, ces agents pourront évidemment lui proposer rapidement un ensemble de décisions permettant à une équipe de prendre des décisions opérationnelles cohérentes si nous supprimons le module d'apport de connaissances. Cette possibilité permettra à l'animateur d'aider des équipes en perdition et ainsi de maintenir la motivation du groupe. Elle passera aussi le message « c'était possible! ». Il est intéressant de prouver que, sur ce type d'outils pédagogiques, des décisions cohérentes amènent à des résultats cohérents. L'inverse est malheureusement aussi vérifié. La malchance ou l'agres-

sivité des concurrents ne peut figurer comme une excuse tangible et durable. La force de réaction d'une équipe à l'adversité doit être mise en avant.

L'exploitation de la connaissance seule pourrait naturellement informer l'animateur sur la stratégie des équipes. Une représentation graphique des positionnements et stratégies en temps réel offrirait assurément un excellent support de connaissance. La projection de ce support offrirait également aux participants une vision de leur action et donc naturellement un recul par effet miroir nécessaire à l'apprentissage pérenne.

Pour le participant, outre une représentation de sa stratégie, une aide à des moments importants du jeu semble opportune. Nous pouvons par exemple citer comme moments clés : l'investissement dans de nouveaux outils de production, la mise en place de moyens de distribution... Ces étapes intègrent naturellement une augmentation par pallier des coûts fixes de l'entreprise. Ces coûts fixes ne sont pas toujours couverts par la nouvelle marge sur coûts variables générée pour la nouvelle activité. La société se retrouve alors en risque financier. Une minimisation du risque pourrait amener les équipes à une croissance plus sereine car mieux connue et donc mieux maîtrisée.

La fourniture des décisions probables des concurrents serait aussi une très bonne information. Elle permettrait un positionnement plus précis et augmenterait la réactivité des opposants.

L'analyse des comportements

Dans les expérimentations réalisées, nous avons exploité des simulations déjà jouées. Ceci nous permettait de construire notre raisonnement et d'entériner les conclusions. Cependant, il serait intéressant d'observer le comportement de joueurs réels qui seraient confrontés aux équipes « optimisées ».

Si l'on transpose cette configuration à la vie économique réelle, nous serions dans la situation de concurrents bien installés sur leur marché qui voient une entreprise extérieure nouvelle, disposant de ressources proches des leurs, s'installer sur le marché. Les réactions à la concurrence feraient l'objet d'analyses stratégiques et comportementales audacieuses. Y aurait-il alliance entre les anciens opérateurs ou plutôt une lutte isolée, entreprise par entreprise ?

Perspectives sur le développement de la méthode

Prédire le vainqueur ?

Depuis le premier chapitre, nous avons annoncé que nous ne souhaitons pas prédire le vainqueur de nos sessions. Ce n'était pas notre objectif. Cependant, nous disposons désormais d'une base de connaissances que nous avons traitée et classée. Par conséquent, nous pourrions y ajouter, pour chaque tour de jeu, le rang atteint par une équipe.

Il serait enrichissant d'imaginer un traitement statistique complémentaire qui offrirait les chances de succès d'une équipe au regard de son parcours sur les arbres d'anticipation.

Quid du scénario ?

Dans notre cas, nous nous sommes focalisés sur un scénario de marché spécifique. Celui-ci nous offrait une base de connaissances exploitable et donc une bonne attractivité pour nos essais. Cependant, nous serions curieux d'identifier l'impact qu'aurait un changement de scénario sur le comportement des agents.

La dimension d'optimisation interne de l'entreprise s'adapterait sans difficulté car seuls les paramètres évolueraient, les règles du jeu resteraient identiques. Par contre, l'analyse de marché demanderait un nouveau travail en profondeur. Comment exploiter les mêmes historiques si nous modifions les événements et donc les réactions des concurrents?

Enrichissement de la base de connaissances et traitement dynamique

L'exploitation des simulateurs en ligne permet aujourd'hui de récupérer facilement et rapidement les décisions des joueurs en cours de partie. D'ailleurs, certains sites de jeux en ligne massivement multijoueurs proposent déjà cette fonctionnalité aux personnes souhaitant exploiter ces données. Il serait logique d'extraire sur l'ensemble des simulations en cours de traitement l'ensemble des décisions des participants, leurs scores, leurs résultats afin d'enrichir, ou de créer en fonction du simulateur, en dynamique la base de connaissances. Cette extension impliquera, à fréquence régulière, le lancement d'opérations de retraitement des données permettant d'ajuster et surtout d'affiner la phase d'anticipation.

Bibliographie

- ABELL, D. F. 1980, *Defining the business : The starting point of strategic planning*, Prentice-Hall Englewood Cliffs, NJ. ix, 26
- ALVAREZ, J. 2007, *Du jeu vidéo au serious game : approches culturelle, pragmatique et formelle*, thèse de doctorat, Toulouse 2. 6
- AMATO, E. A. 2007, «Vers une instrumentalisation communicationnelle des jeux vidéo : quelles formes de séduction idéologique ou publicitaire?», dans *Colloque international EUTIC 2007 : " Enjeux et Usages des TIC"*, 7-10 novembre 2007. 6
- AXELROD, R. M. 2006, *The evolution of cooperation*, Basic books. 79
- BAKER, E. L. et H. F. O'NEIL. 2002, «Measuring problem solving in computer environments : current and future states», *Computers in Human Behavior*, vol. 18, n° 6, p. 609–622. 5
- BARDOS, M. 2001, «Développements récents de la méthode des scores de la banque de france», *Bulletin de la Banque de France*, , n° 90, p. 73–92. 16
- BARDOS, M. et collab.. 2005, «Les scores de la banque de france : leur développement, leurs applications, leur maintenance», *Bulletin de la Banque de France*, vol. 144, p. 63–73. 16
- BARTH, I. et I. GÉNIAUX. 2010, «Former les futurs managers à des compétences qui n'existent pas : les jeux de simulation de gestion comme vecteur d'apprentissage», *Management & Avenir*, , n° 6, p. 316–339. 7
- BEN-DAVID, S. et M. ACKERMAN. 2009, «Measures of clustering quality : A working set of axioms for clustering», dans *Advances in neural information processing systems*, p. 121–128. 92
- BEREITER, C. 2005, *Education and mind in the knowledge age*, Routledge. 5
- BLUM, C. et A. ROLI. 2003, «Metaheuristics in combinatorial optimization : Overview and conceptual comparison», *ACM Computing Surveys (CSUR)*, vol. 35, n° 3, p. 268–308. 59
- BOUSSAÏD, I., J. LEPAGNOT et P. SIARRY. 2013, «A survey on optimization metaheuristics», *Information Sciences*, vol. 237, p. 82–117. 59
- BREIMAN, L., J. FRIEDMAN, C. J. STONE et R. A. OLSHEN. 1984, *Classification and regression trees*, CRC press. 102
- BRESSY, G. 2014, *Management et économie des entreprises*, Sirey Dalloz, Paris, ISBN 978-2247136711. xiii, 24
- CALIŃSKI, T. et J. HARABASZ. 1974, «A dendrite method for cluster analysis», *Communications in Statistics-theory and Methods*, vol. 3, n° 1, p. 1–27. 93
- COOPER, D., M. EZZAMEL et S. QU. 2016, «Popularizing a management accounting idea : The case of the balanced scorecard», *Available at SSRN*. 31
- COVER, T. et P. HART. 1967, «Nearest neighbor pattern classification», *IEEE transactions on information theory*, vol. 13, n° 1, p. 21–27. 103

- DAVIES, D. L. et D. W. BOULDIN. 1979, «A cluster separation measure», *IEEE transactions on pattern analysis and machine intelligence*, n° 2, p. 224–227. [94](#)
- DEMIL, B., X. LECOCQ et V. WARNIER. 2013, *Stratégie et business models*, Pearson Education France. [ix](#), [24](#)
- DHAENENS, C. et L. JOURDAN. 2016, *Metaheuristics for Big Data*, John Wiley & Sons, Inc., ISBN 9781119347569, 147–158 p., doi :10.1002/9781119347569.ch8. URL <http://dx.doi.org/10.1002/9781119347569.ch8>. [100](#)
- DISNEY, S. M. et D. R. TOWILL. 2003, «The effect of vendor managed inventory (vmi) dynamics on the bullwhip effect in supply chains», *International journal of production economics*, vol. 85, n° 2, p. 199–215. [9](#)
- DRACHEN, A., C. THURAU, R. SIFA et C. BAUCKHAGE. 2014, «A comparison of methods for player clustering via behavioral telemetry», *CoRR*, vol. abs/1407.3950. URL <http://arxiv.org/abs/1407.3950>. [21](#), [82](#), [89](#), [96](#)
- DRUCKER, P. F. et P. F. DRUCKER. 1994, *Post-capitalist society*, Routledge. [5](#)
- DUBRULLE, L. et D. JOURDAIN. 2013, *Comptabilité analytique de gestion-6ème édition*, Dunod. [27](#), [29](#)
- DUFOURNY, S. et C. DHAENENS. 2013, «Modèle d'optimisation dirigé par la production et la distribution dans une simulation d'entreprise», dans *ROADéf 2013, Congrès de la Société française de Recherche Opérationnelle et d'Aide à la Décision*. [42](#)
- DUFOURNY, S. et C. DHAENENS. 2015, «An original hybrid approach to optimize economic decisions in a business game», dans *Proceedings of the Metaheuristics International Conference(MIC'15)*. [40](#), [57](#)
- DUFOURNY, S. et C. DHAENENS. 2016, «Optimization of business game decisions», dans *8th IFAC Conference on Manufacturing Modelling, Management & Control (MIM 2016)*. [43](#)
- EASLEY, D. et J. KLEINBERG. 2010, *Networks, crowds, and markets : Reasoning about a highly connected world*, Cambridge University Press. [78](#)
- EIKLAND, K. 2013. URL <http://http://lpsolve.sourceforge.net/5.5>. [53](#)
- ETTENSON, R., E. CONRADO et J. KNOWLES. 2013, «Rethinking the 4 p's», *Harvard business review*, vol. 91, n° 1, p. 26–27. [25](#)
- FORAY, D. 2010, *L'économie de la connaissance*, La découverte. [5](#)
- FORGY, E. W. 1965, «Cluster analysis of multivariate data : efficiency versus interpretability of classifications», *Biometrics*, vol. 21, p. 768–769. [90](#)
- FORRESTER, J. et K. RIEMER. 2007. URL <http://www.beergame.org>. [ix](#), [10](#)
- GARÈNE, M. 1992, «Donnant donnant : Une théorie de comportement coopératif», *The Evolution of Cooperation*. [80](#)
- GARIBALDI, G. 2007, *Analyse strategique*, Editions Eyrolles. [ix](#), [27](#)

- GENDREAU, M. et J.-Y. POTVIN. 2010, *Handbook of metaheuristics*, vol. 2, Springer. 60
- GLOVER, F. 1989, «Tabu search-part i», *ORSA Journal on computing*, vol. 1, n° 3, p. 190–206. 62
- GLOVER, F. et M. LAGUNA. 2013, *Tabu Search**, Springer. 62
- GOODWIN, J. S. et S. G. FRANKLIN. 1994, «The beer distribution game : using simulation to teach systems thinking», *Journal of Management Development*, vol. 13, n° 8, p. 7–15. 11
- HAN, J., J. PEI et M. KAMBER. 2011, *Data mining : concepts and techniques*, Elsevier. 89
- HAO, Y. O., D. SUNIL et P. MARK. 2015, «Player behavior and optimal team composition for online multiplayer games», *CoRR*, vol. abs/1503.02230. URL <http://arxiv.org/abs/1503.02230>. 21, 82, 89, 96
- HARGREAVES, A. 2003, *Teaching in the knowledge society : Education in the age of insecurity*, Teachers College Press. 5
- HARSANYI, J. C. 2004, «Games with incomplete information played by “bayesian” players, i–iii : part i. the basic model&», *Management science*, vol. 50, n° 12_supplement, p. 1804–1817. 81
- HAUSMAN, J. et D. MCFADDEN. 1984, «Specification tests for the multinomial logit model», *Econometrica : Journal of the Econometric Society*, p. 1219–1240. 102
- HENSHER, D. A. et W. H. GREENE. 2003, «The mixed logit model : the state of practice», *Transportation*, vol. 30, n° 2, p. 133–176. 102
- HILL, C. W. 1988, «Differentiation versus low cost or differentiation and low cost : a contingency framework», *Academy of Management Review*, vol. 13, n° 3, p. 401–412. 99
- IGLEWICZ, B. et D. HOAGLIN. 1993, «Volume 16 : how to detect and handle outliers», *The ASQC Basic Reference in Quality Control : Statistical Technique*. 88
- JOHNSON, D. W., R. T. JOHNSON et K. A. SMITH. 1998, *Active learning : Cooperation in the college classroom*, ERIC. 6
- JOHNSON, G. 2014, *Stratégique*, Pearson, Montreuil, ISBN 978-2326000421. 24
- KIMBROUGH, S. O., D.-J. WU et F. ZHONG. 2002, «Computers play the beer game : can artificial agents manage supply chains?», *Decision support systems*, vol. 33, n° 3, p. 323–333. 11
- KRIZ, W. C. 2008, «A systemic-constructivist approach to the facilitation and debriefing of simulations and games», *Simulation & Gaming*. 6
- LEHTINEN, E. 2002, «Developing models for distributed problem-based learning : Theoretical and methodological reflection», *Distance Education*, vol. 23, n° 1, p. 109–117. 5
- LOH, W.-Y. 2011, «Classification and regression trees», *Wiley Interdisciplinary Reviews : Data Mining and Knowledge Discovery*, vol. 1, n° 1, p. 14–23. 108

- LOURENÇO, H. R., O. C. MARTIN et T. STÜTZLE. 2010, «Iterated local search : Framework and applications», dans *Handbook of Metaheuristics*, Springer, p. 363–397. [62](#)
- MACHUCA, J. A. 2000, «Transparent-box business simulators : An aid to manage the complexity of organizations», *Simulation & Gaming*, vol. 31, n° 2, p. 230–239. [5](#)
- MACQUEEN, J. et collab.. 1967, «Some methods for classification and analysis of multivariate observations», dans *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol. 1, Oakland, CA, USA., p. 281–297. [91](#)
- MANNING, C., P. RAGHAVAN et H. SCHÜTZE. 2008, «Vector space classification», *Introduction to Information Retrieval*. [103](#)
- MATLAB. 2010, *version 7.10.0 (R2010a)*, The MathWorks Inc., Natick, Massachusetts. [90](#), [92](#)
- MCCARTHY, E. J. et W. D. PERREAULT. 1993, *Basic marketing : a global-managerial approach*, Irwin Homewood, IL. [25](#)
- MEIER, O. 2015, *Diagnostic stratégique-4e éd. : Évaluer la compétitivité de l'entreprise*, Dunod. [24](#)
- MOUAHEB, H., A. FAHLI, M. MOUSSETAD et S. ELJAMALI. 2012, «The serious game : what educational benefits?», *Procedia-Social and Behavioral Sciences*, vol. 46, p. 5502–5508. [6](#)
- MOULIN, H. 1991, *Axioms of cooperative decision making*, 15, Cambridge University Press. [80](#)
- NACCACHE, S., B. MONTREUIL, O. LABARTHE, E. BROTHERTON et S. MELLOULI. 2010, «Simulations participatives du xbeergame : Comparaisons des performances entre joueurs humains et agents décisionnels», dans *Proceedings of the 8th ENIM IFAC International Conference of Modeling and Simulation*, p. 824–833. [64](#)
- NISAN, N., T. ROUGHGARDEN, E. TARDOS et V. V. VAZIRANI. 2007, *Algorithmic game theory*, vol. 1, Cambridge University Press Cambridge. [78](#)
- PECH-GOURG, N. et S. DUFOURNY. 2012, «Optimisation des choix d'investissement, de production, et de sous-traitance dans un simulateur de gestion d'entreprise», dans *ROADéf 2012, Congrès de la Société française de Recherche Opérationnelle et d'Aide à la Décision*. [42](#)
- PIGET, P. 2015, *De la comptabilité de gestion au contrôle de gestion*, Economica, Paris, ISBN 271786766X. [27](#), [29](#)
- PORTER, M. E. et P. DE LAVERGNE. 2003, *L'avantage concurrentiel comment devancer ses concurrents et maintenir son avance*, Dunod, Paris, ISBN 210007394X. [99](#), [123](#), [125](#), [126](#), [128](#)
- PUCHINGER, J. et G. R. RAIDL. 2005, «Combining metaheuristics and exact algorithms in combinatorial optimization : A survey and classification», dans *International Work-Conference on the Interplay Between Natural and Artificial Computation*, Springer, p. 41–53. [59](#)

- QUINLAN, J. R. 2014, *C4. 5 : programs for machine learning*, Elsevier. 102, 108
- RAPOPORT, A. et A. M. CHAMMAH. 1965, *Prisoner's dilemma : A study in conflict and co-operation*, vol. 165, University of Michigan press. 78
- RIEMER, K. 2008, «The beergame in business-to-business ecommerce courses-a teaching report», *BLED 2008 Proceedings*, p. 1. 11
- SAADA, T. 2005, *Comptabilité analytique et contrôle de gestion*, Vuibert, Paris, ISBN 2711795799. 27
- SAWYER, R. 2008, «Optimiser l'apprentissage : Les implications de la recherche en sciences de l'apprentissage», dans *Communication présentée dans le cadre de Conférence internationale OCDE/CERI : Apprendre au XXIe siècle : recherche, innovation et politique. Paris, France.* 5
- SHIMIZU, K. 2009, «Advertising theory and strategies», *Souseisha Book Company. (Japanese).* 25
- SHOHAM, Y. et K. LEYTON-BROWN. 2008, *Multiagent systems : Algorithmic, game-theoretic, and logical foundations*, Cambridge University Press. 78
- STRAT&LOGIC. 2000. URL <http://www.stratelogic.com>. 11, 84
- TALBI, E.-G. 2009, *Metaheuristics : from design to implementation*, vol. 74, John Wiley & Sons. 59
- TUCKER, A. W. 1950, «A two-person dilemma», *Readings in games and information*, p. 7–8. 78
- TUFFÉRY, S. 2012, *Data mining et statistique décisionnelle : l'intelligence des données*, Editions Technip. 101
- VERHULST, P. F. 1845, «Recherches mathématiques sur la loi d'accroissement de la population.», *Nouveaux mémoires de l'académie royale des sciences et belles-lettres de Bruxelles*, vol. 18, p. 14–54. 102
- WARD JR, J. H. 1963, «Hierarchical grouping to optimize an objective function», *Journal of the American statistical association*, vol. 58, n° 301, p. 236–244. 92

Annexe A

Modélisation du département Distribution

A.1 Programme linéaire et contraintes du département Distribution

A.1.1 Notation

Les données du département Distribution sont listées dans le tableau [A.1](#). Ce tableau présente les grandeurs de distribution : capacités de distribution, ventes prévues... Pour mémoire, ces ventes prévisionnelles sont le fruit d'une analyse du marché et de la concurrence (Figure [2.11](#)).

TABLEAU A.1 – Paramètres de distribution

HV	Grande valeur arbitraire (high value)
h, h'	Mode de distribution
p	Produit p
$SalesCapa_{h,z}$	Capacité de vente pour un point de vente de type h et sur la zone z
$Price_{p,z,n}$	Prix du produit p au tour n sur la zone z
$Sales_{p,z,n}$	Ventes prévues du produit p au tour n sur la zone z
$D_{z,h}$	Ensemble des points de vente pour la zone z de type h

Afin de distribuer les ventes prévisionnelles, il faut définir le nombre de points de vente à mettre en place. Il existe classiquement dans la simulation UpStrat deux type de points de vente. Le premier est représenté par des agents commerciaux qui sont rémunérés en fonction du chiffre d'affaires généré. Lorsque l'entreprise se sépare de ses agents commerciaux, des coûts représentant des indemnités de rupture de contrat sont à payer. Les succursales, propriétés de l'entreprise, représentent le second mode de distribution. Elles imposent un investissement et génèrent des coûts fixes réguliers.

Le tableau [A.2](#) indique les variables de décisions de distribution, pour chaque zone z au tour n .

Afin de calculer la capacité de distribution, chaque point de vente d est décrit par l'ensemble de variables reprises dans le tableau [A.3](#) au tour n :

Le paragraphe suivant présente les contraintes de distribution à remplir pour chaque hypothèse considérée (début de construction d'une solution). La fonction objectif sera

TABLEAU A.2 – Variables de distribution pour une zone z au tour n

$NbDist_{z,n}$	nombre de points de vente pour la zone z
$IsType_{z,n,h}$	= 1 si le mode de distribution est de type h pour la zone z (0 sinon)
$isChanged_{z,n}$	= 1 si le mode de distribution est modifié pour la zone z (0 sinon)

TABLEAU A.3 – Variable des points de vente pour le tour n

$isBought_{d,n}$	= 1 si le point de vente d est acheté =(0 sinon)
$isTransfer_{d,n}$	= 1 si le point de vente d est vendu =(0 sinon)
$isUsed_{d,n}$	= 1 si le point de vente d est exploité =(0 sinon)
$CapitalLoss_{d,n}$	moins value du point de vente d
$isUsed_{d,n}$	= 1, si le point de vente d est utilisé(0 sinon)
$isAged_{d,n,i}$	= 1 si le point de vente d est agé de i (0 sinon)

présentée en fin de paragraphe.

A.1.2 Contraintes et modèle de programmation linéaire

Le principal objectif du département distribution est d'être en capacité de distribuer la demande client (A.1). Ainsi, les entreprises doivent présenter une capacité de distribution au moins équivalente aux ventes estimées valorisées par zone z (soit le chiffre d'affaires). Cette contrainte se traduit de la manière suivante :

$$\forall z, \forall n, \sum_p Price_{p,z,n} * Sales_{p,z,n} \leq SalesCapa_{z,n} \quad (A.1)$$

Avant de calculer les capacités de distribution par zone, il faut évaluer $isUsed_{d,n}$ qui indique que le point de vente d est actif au cycle n . Pour cela, nous rappelons que le type de distributeur est exclusif par zone. Cette contrainte s'exprime par l'équation (A.2).

$$\forall n, \forall z, \sum_h IsType_{z,n,h} = 1 \quad (A.2)$$

Un booléen, $isChanged_{z,n}$ est utilisé afin de détecter si le type de point de vente a été modifié (Contraintes (A.3) et (A.4)).

$$\forall z, \forall h, \forall n \geq 2, \quad isChanged_{z,n} \geq IsType_{z,n-1,h} - IsType_{z,n,h} \quad (A.3)$$

$$\forall z, \forall h, \forall n \geq 2, \quad isChanged_{z,n} \geq IsType_{z,n,h} - IsType_{z,n-1,h} \quad (A.4)$$

L'utilisation d'un point de vente dépend naturellement de son achat ($isBought$) et de son éventuelle vente ($isTransfer$).

$$\forall d, \forall n, \quad IsUsed_{d,n} = \sum_{j=1}^n isBought_{d,j} - \sum_{j=1}^n isTransfer_{d,j} \quad (A.5)$$

Nous sommes donc en capacité d'évaluer le nombre de distributeurs par zone au moyen des équations (A.6) et (A.7).

$$\forall z, \forall h, \quad NbDist_{z,n,h} = \sum_{D_{z,h}, i \leq n} isBought_{d,i} - \sum_{D_z, i \leq n} isTransfer_{d,i} \quad (A.6)$$

$$\forall n, \forall z, \forall h, \quad NbDist_{z,n} \geq NbDist_{z,n,h} - HV * \sum_{h' \neq h} IsType_{z,n,h'} \quad (A.7)$$

Chaque point de vente ne peut être vendu qu'une seule fois ($n \in [n_0, \dots, n_0 + k]$) :

$$\forall d, \quad \sum_{n=1}^{n_0+k} isTransfer_{d,j} \leq 1 \quad (A.8)$$

Chaque point de vente ne peut être acheté qu'une seule fois ($n \in [n_0, \dots, n_0 + k]$) :

$$\forall d, \quad \sum_{j=1}^{n_0+k} isBought_{d,j} \leq 1 \quad (A.9)$$

Afin d'évaluer le coût d'une cession d'un point de vente (moins-value), nous devons évaluer l'âge du bien présenté par la variable *isAged*. Toutes les valeurs de *isAged* prennent 0 si le point de vente est vendu.

$$\forall d, \forall n, \quad \sum_{i=1}^{n_0+k} isAged_{d,n,i} \leq 1 - isTransfer_{d,n} \quad (A.10)$$

$$\forall d, \forall i \forall j, \forall (i - j + n_0 > 0) \quad (A.11)$$

$$IsAged_{d,i,j} = isBought_{d,i-j+n_0} - \sum_{k=1}^{i+n_0} isTransfer_{d,k}$$

$$\forall d, \forall i \forall j, \forall (i - j + n_0 \leq 0) \quad IsAged_{d,i,j} = 0 \quad (A.12)$$

La moins value du point de vente est donc calculé par l'équation suivante :

$$\forall n, \forall d, \quad CapitalLoss_{d,n} \geq \sum_{i=1}^{10} Cst_i * isAged_{d,n,i} - HV * (1 - isTransfer_{d,n}) \quad (A.13)$$

La capacité de distribution dépend du critère d'activité (A.15). Elle est donc calculée au moyen des équations suivantes :

$$\forall z, \forall n, \quad SalesCapa_{z,n} \leq \sum_{D_z} SalesCapa_{d,n} \quad (A.14)$$

$$\forall d, \forall n, \forall h, \quad SalesCapa_{d,n} \leq CstDist_{z,h} * isUsed_{d,n} \quad (A.15)$$

A.1.3 La fonction objectif

Le but de la fonction objectif est de minimiser l'ensemble des coûts de distribution. Les coûts de distribution sont listés dans le tableau A.4.

TABEAU A.4 – Coûts intégrés dans le programme linéaire de Distribution

<i>Ct_Dismiss</i>	Coût de rupture de contrat des points de vente
<i>Ct_Amort</i>	Amortissement des points de vente
<i>Ct_Fixed</i>	Coût fixe de distribution
<i>Ct_Variable</i>	Coût variable de distribution
<i>Capital_Loss</i>	Moins value des cessions de points de vente

La fonction objectif peut alors être exprimée par :

$$\text{Min} \left(\sum_{n,z} \begin{pmatrix} CT_Dismiss_{z,n} + \\ CT_Amort_{z,n} + \\ CT_Fixed_{z,n} + \\ CT_Variable_{z,n} \end{pmatrix} + \sum_n Capital_Loss_n \right) \quad (\text{A.16})$$

Chaque coût est obtenu par les équations suivantes :

Le **coût de rupture de contrat** est obtenu par l'équation (A.17).

$$\forall n, \forall z, \quad Ct_Dismiss_{z,n} = \sum_{D_z, i=n} isTransf_{d,i} * CstRupture_z \quad (\text{A.17})$$

avec $CstRupture_z$ le coût de rupture de contrat fixé en paramétrage de la zone z .

Le coût correspondant à la **dotation aux amortissements** est donné par l'équation (A.18).

$$\forall n, \forall z, \forall h, \quad Ct_Amort_{z,n} \geq NbPt_{z,n} * CstAmort_{z,h} - HV * \sum_{h' \neq h} IsType_{z,n,h'} \quad (\text{A.18})$$

avec $CstAmort_{z,h}$ la dotation aux amortissements d'un point de vente de type h sur la zone z . L'amortissement est linéaire donc cette valeur est identique pour les cycles considérés à partir de sa date d'achat.

Le **coût fixe** est évalué par l'équation (A.19).

$$\forall n, \forall z, \forall h, \quad Ct_fixed_{z,n} \geq NbDist_{type_h} * CstFixe_{z,h} - HV * \sum_{h' \neq h} IsType_{z,n,h'} \quad (\text{A.19})$$

avec $CstFixe_{z,h}$ le paramètre de coût fixe pour un point de vente de type h sur la zone z .

Le **coût variable** est évalué par l'équation (A.20).

$$\forall n, \forall z, \forall h, \quad Ct_variable_{z,n} \geq \sum_{p,h} Sales_{p,z,n} * Price_{p,z,n} * Commission_{z,h} * isType_{z,h} \quad (\text{A.20})$$

avec $Commission_z$ le taux de commission reversé au point de vente sur la zone z s'il est de type h .

La **moins value de cession** est évaluée par l'équation (A.21).

$$\forall n, \forall z, \forall h, \quad Capital_Loss_{z,n,h} = \sum_{d \in D_z} Capital_Loss_d \quad (\text{A.21})$$

$$\forall n, \forall z, \forall h, \quad Capital_Loss_{d,n} \geq Capital_Loss_{d,n,h} - HV * \sum_{h' \neq h} IsType_{d,n,h'} \quad (\text{A.22})$$

Donc :

$$\forall n, \quad Capital_Loss_n = \sum_{d \in D} Capital_Loss_{d,n} \quad (\text{A.23})$$

Optimisation du modèle

De la même manière que pour le programme linéaire du département Production, nous avons souhaité optimiser le modèle. Ainsi, pour éviter la génération de solutions équivalentes avec différents indices de points de vente, nous ajoutons la contrainte (A.24). Nous forçons ainsi l'optimisation à utiliser les points de vente disponibles de plus faible indice (A.25).

$$\forall n, \forall z, \forall h, \quad isType_{z,n,h} * HV \geq \sum_{d \in D_{z,h}} \left(\sum_{i \leq n} isBought_{d,i} - \sum_{i \leq n} isTransfer_{d,i} \right) \quad (A.24)$$

$$\forall d, \forall n, \quad isBought_{d,n} \leq \sum_{i \leq n} isBought_{d-1,i} \quad (A.25)$$

La contrainte (A.26) supprime toutes les solutions dupliquées. De plus, elle supprime de la résolution les points de vente dont le type ne correspond pas au type actuel.

$$\forall n, \forall z, \quad \sum_{d \in D_{z,h}} \left(\sum_{i \leq n-1} isBought_{d,i} - \sum_{i \leq n} isTransfer_{d,i} \right) \leq (1 - isChanged_{z,n}) * HV \quad (A.26)$$

