



HAL
open science

La modélisation de l'indice CAC 40 avec le modèle basé agents

Nan Lu

► **To cite this version:**

Nan Lu. La modélisation de l'indice CAC 40 avec le modèle basé agents. Economies et finances. Université Paris-Est, 2018. Français. NNT : 2018PESC0004 . tel-01950289

HAL Id: tel-01950289

<https://theses.hal.science/tel-01950289>

Submitted on 10 Dec 2018

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

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

UNIVERSITÉ — — PARIS-EST

THÈSE

En vue de l'obtention du
DOCTORAT DE L'UNIVERSITÉ PARIS EST

Délivré par : *l'Université Paris Est*

Présentée et soutenue le *13/03/2018* par :

Nan LU

La modélisation de l'indice CAC 40 avec un modèle basé agents

JURY

FRANÇOIS LEGENDRE
MÉLIKA BEN SALEM
JEAN-DANIEL KANT
ANNICK VIGNES

Professeur de l'Université Paris Est
Professeur de l'Université Paris Est
Maître de Conférences de Sorbonne Université
Directrice de recherche en économie de
l'École des Ponts ParisTech

Membre de Jury
Membre de Jury
Membre de Jury
Membre de Jury

École doctorale et spécialité :

OMI : Organisations, Marchés, Institutions

Unité de Recherche :

*ÉRUDITE (Équipe de Recherche sur l'Utilisation des Données Individuelles en lien avec la
Théorie Économique, EA N437)*

Directeur de Thèse :

François LEGENDRE

Rapporteurs :

Annick VIGNES et Jean-Daniel KANT

The logo for Université Paris-Est Créteil (UPEC) features the letters 'UPEC' in a large, bold, grey sans-serif font. A pink diagonal bar is positioned behind the letters, extending from the bottom left towards the top right.

UNIVERSITÉ PARIS-EST CRÉTEIL

Remerciements

En premier lieu, je tiens à remercier tout particulièrement mon directeur de thèse, Monsieur FRANÇOIS LEGENDRE, Directeur du Département de Science Économique à l'Université Paris Est Créteil, pour la confiance qu'il m'a accordée en acceptant d'encadrer ce travail doctoral, pour ses multiples conseils et pour toutes les heures qu'il a consacrées à diriger cette recherche tout au long de ces cinq dernières années. Sa capacité d'analyse des problèmes, ses très nombreuses connaissances et son enthousiasme m'ont permis de progresser et m'ont montré que le monde de la recherche pouvait être un univers passionnant. J'aimerais également lui dire à quel point j'ai apprécié sa grande disponibilité, surtout pendant les vacances, et son respect sans faille des délais serrés de relecture des documents et des programmes que je lui ai envoyés. Enfin, j'ai été extrêmement sensible à ses qualités humaines d'écoute et de compréhension.

Mes remerciements vont également à remercier Madame ANNICK VIGNES, Directrice de recherche en économie de l'École des Ponts ParisTech, et à Monsieur JEAN-DANIEL KANT, Maître de Conférences de Sorbonne Université pour l'honneur qu'ils m'ont fait en acceptant d'être rapporteurs de mon travail et pour leurs remarques qui ont permis d'améliorer la qualité de cette thèse. Je tiens à leur exprimer toute ma reconnaissance pour avoir accepté de participer à ce jury.

Je remercie aussi Madame MELIKA BEN SALEM, Professeur de l'Université Paris Est, qui a bien voulu accepter de faire partie du jury.

J'adresse mes remerciements les plus chaleureusement à tous les membres du Laboratoire de l'ÉRUDITE, SANDRINE, ERCI, SYLVAIN, ADEM, FAREZ, pour leur aide et leur bonne humeur. Nous avons ensemble partagé de bons moments lors des pauses café.

Un grand merci à mon collègue et ami, ADEL ARAB, correcteur des fautes d'orthographe de cette thèse.

Je remercie aussi Madame RÉGINE PAJOT, pour la correction de cette thèse.

Enfin, je remercie ma chère épouse, BAIHUI HAN, pour son soutien quotidien indéfectible et son enthousiasme contagieux à l'égard de mes travaux comme dans la vie en général. Notre couple a grandi en même temps que mon projet scientifique. Notre fils LOUKA et notre fille EMMA nous apportent beaucoup de joie.

*À ma mère et mon père,
ils sauront pourquoi ...*

Table des matières

Introduction générale	8
1 Une revue de l’histoire du modèle <i>basé agents</i>	11
1.1 Introduction	11
1.2 Naissance du modèle <i>basé agents</i>	13
1.3 Critiques des théories économiques traditionnelles	15
1.4 Essor des sciences des systèmes complexes	18
1.5 Recherche sur le processus d’apprentissage et évolution du comportement <i>basé agents</i>	20
1.5.1 Brève histoire du développement et de la recherche du modèle <i>basé</i> <i>agents</i>	21
1.5.2 Premières recherches sur le modèle <i>basé agents</i> en finance	25
1.6 Le modèle <i>basé agents</i> et la physique	27
2 Les méthodes et les procédures générales de la modélisation du modèle <i>basé</i> <i>agents</i>	37

2.1	Les technologies clés de la modélisation du modèle <i>basé agents</i>	39
2.1.1	Le modèle d'agent	40
2.1.2	Les interactions entre agents	46
2.2	L'analyse des résultats de l'émergence du modèle <i>basé agents</i>	53
2.2.1	Analyse des règles individuelles	53
2.2.2	Analyse statistique des séries temporelles	54
2.2.3	Analyse de l'animation graphique	55
2.2.4	Autres méthodes	55
2.3	Critiques du modèle <i>basé agents</i>	57
3	Anomalies sur les marchés financiers	61
3.1	Introduction	61
3.2	Queue épaisse	66
3.2.1	Les données	67
3.2.2	La distribution des rendements du <i>CAC 40</i>	68
3.3	Mémoire à long terme et volatilité clustering	70
3.3.1	Définition de la mémoire longue	71
3.3.2	Analyse de la mémoire longue du <i>CAC 40</i>	72
3.4	Autres anomalies sur le marché financier	78
3.4.1	Les effets sur la petite entreprise	78

3.4.2	Les effets de calendrier	80
3.5	La mesure de reproduction des anomalies	81
3.5.1	Le <i>Moving Block Bootstrap</i>	83
3.5.2	La fonction objective $f(\theta)$ pour la simulation	90
4	Notre propre modèle basé agents et nos résultats	102
4.1	Notre modèle <i>basé agents</i>	104
4.1.1	Agents positionnés dans un réseau	104
4.1.2	Mécanisme de la prise de décision	106
4.1.3	Formation du cours boursier	111
4.1.4	Transactions pour chaque agent	112
4.1.5	Premiers résultats et problèmes rencontrés	116
4.2	Amélioration du modèle et des résultats : le mimétisme des anticipations permet-il de reproduire les anomalies observées sur les marchés finan- ciers?	118
4.3	Le modèle de percolation	122
4.3.1	Le modèle de CONT et BOUCHAUD (2000)	122
4.3.2	Deux développements récents des <i>modèles basés agents</i> en finance	125
4.4	Un modèle simple	131
4.5	Estimation du modèle et résultats	136

4.5.1	Quels moments retenir?	137
4.5.2	Les résultats	140
4.6	Conclusion	149
Conclusion générale et perspectives		151
Bibliographie		154
Annexes		
A	Programmes	i
A.1	Calculs des statistiques du <i>CAC 40</i>	i
A.2	Mémoire longue	iii
A.3	Nombre de cluster	v
A.4	Estimation de la variance de l'estimateur de ρ_1 pour un <i>AR1</i> par simulation et par bootstrap	vi
A.5	Réseau	viii
A.6	Calcul de ω et ω^{-1}	ix
A.7	Relation entre le ω et la taille du bloc	xi
A.8	Programme principal	xii

Introduction générale

Sur le marché financier, il existe de nombreuses anomalies que les théories économiques traditionnelles ne peuvent bien expliquer. Cependant, en tant que théorie économique émergente, le modèle *basé agents* (*Agent-based computational economics, ACE*) est capable de reproduire certaines des propriétés statistiques des rendements observés sur les marchés boursiers réels.

Dans cette thèse, je reviendrai d'abord brièvement sur l'histoire du modèle *basé agents* et indiquerai les critiques de cette théorie. Ensuite, je présenterai les méthodes et les procédures générales de la modélisation du modèle *basé agents*. Je propose également dans cette recherche, un modèle très simple permettant de reproduire les deux principales « anomalies » que l'on observe sur les marchés financiers : d'un côté, une distribution des rendements dont les queues seraient trop épaisses, de l'autre, une volatilité irrégulière dans le temps du cours des actifs financiers. Je voudrais notamment montrer, en utilisant un modèle *basé agents*, qu'une formation mimétique des anticipations est en mesure de rendre compte de ces deux caractéristiques.

Les causes du scepticisme vis-à-vis du recours aux modèles *basés agents* sont discutées. Ces causes ne sont pas sans fondement : ces modèles sont bien souvent sur-paramétrés (dans leur ambition de décrire de manière réaliste les comportements), étant calibrés et mal validés. J'envisage de lever ce scepticisme en exposant un modèle simple, estimé à l'aide d'une méthode économétrique.

Je me propose de reproduire les évolutions de l'indice *CAC 40* qui sont notamment caractérisées pas une distribution leptokurtique et par une volatilité par grappes.

Pour ce faire, je m'appuie sur le modèle dit de percolation développé par CONT et BOUCHAUD (2000), mais en m'écartant sensiblement des développements ultérieurs proposés, par exemple, par IORI (2002) ou FABRETTI (2013) qui cherchent, en complexifiant le modèle, à mieux reproduire les évolutions observées. Le modèle développé par IORI (2002) est seulement simulé, demeurant quelque peu en forme de « boîte noire ». En revanche, celui proposé par FABRETTI (2013) est estimé par une méthode de moments, mais il semble sous-identifié en raison de son très grand nombre de paramètres.

Je paramètre le modèle en distinguant trois sources d'information qui contribueraient à fixer les décisions d'achat ou de vente de l'actif financier : une source idiosyncratique, une source liée au voisinage de chaque agent et, enfin, une source proprement macroéconomique. La source liée au voisinage, associée au comportement mimétique, pourrait expliquer la distribution leptokurtique ; la source macroéconomique contribue à la volatilité par grappe.

Je discute ensuite du critère à retenir afin d'estimer le modèle par la méthode des moments. Je montre que le critère qui repose sur la matrice de variance-covariance des moments que l'on cherche à reproduire n'est pas nécessairement le plus pertinent, même s'il semble statistiquement fondé. Par une méthode de *moving block bootstrap*, je calcule la matrice de variance-covariance des moments d'intérêt pour le cours journalier du *CAC 40* depuis sa création en 1988, et je montre que ce critère ne donne pas une place suffisante, par exemple, au kurtosis.

Enfin, je présente les premiers résultats en prenant un algorithme évolutif *CMA-ES* (*Covariance Matrix Adaptation Evolution Strategy*) afin de régler le problème d'optimisation continu.

Tous mes modèles sont réalisés à l'aide du compilateur C++.

Le premier Chapitre est consacré à une brève revue de l'histoire du modèle *basé agents*. Il s'agira de remonter à l'origine de ce modèle et d'introduire son développement pendant ces dernières années. La relation entre le modèle *basé agents* et la physique sera expliquée. Le chapitre suivant présente les techniques et les méthodes de la modélisation du modèle *basé agents*. Le Chapitre 3 analyse les anomalies sur l'indice CAC 40. Quant au Chapitre 4, il traite des critiques sur le modèle *basé agents* et son défi. Le modèle *basé agents* sera estimé par rapport au modèle économique classique. D'autres idées concernant la simulation avec l'ordinateur seront également proposées.

Chapitre 1

Une revue de l'histoire du modèle *basé agents*

The failure to predict the financial crisis was a “Michael Fish” moment for economists.

Andy Haldane, économiste en chef à la Banque d'Angleterre

1.1 Introduction

Jeudi 5 janvier 2017, ANDREW HALDANE est apparu sur la scène de l'actualité, ayant comparé l'économiste au prévisionniste. Bien évidemment, ce n'était pas une comparaison flatteuse pour l'économiste. Il a indiqué que le prévisionniste britannique, MICHAEL FISH, avait déclaré le 15 octobre 1987, lors d'une prévision : «*Earlier on today, apparently, a woman rang the BBC and said she heard there was a hurricane on the way... well, if you're watching, don't worry, there isn't!*». Quelques heures plus tard, une grande tempête frappe le sud-est de l'Angleterre et la France, tuant respectivement 19 et 15 personnes.

La leçon de cet événement est que les économistes devraient poursuivre leurs efforts

afin d'établir des prévisions. Il est clair que la tâche des météorologues est ardue, mais ils assurent bien leur travail grâce aux 50 millions de mesures météorologiques quotidiennes et à la simulation d'un ordinateur super puissant. Peut-être les prévisions économiques devraient-elles également suivre l'exemple de cette approche. Ainsi, peut-on utiliser des ordinateurs pour simuler l'économie réelle ? La réponse est affirmative.

En 2008, la Reine ÉLISABETH II a interrogé un groupe d'économistes au sujet de la crise financière lors d'une visite à la London School of Economics : «*Why did no one see it coming ?*» Pourquoi personne ne l'avait vu venir ? De nombreux éco-physiciens et *Santa Fe Institute (SFI)* disent que la réponse se trouve dans le modèle *Agent-based computational Economics*.

Le 30 octobre 2008, le français JEAN-PHILIPPE BOUCHAUD, cofondateur et président de la société *Capital Fund Management (CFM)*, et professeur à l'école polytechnique de Paris, a publié un article «*Economics need a scientific revolution* » dans *Nature*¹ (BOUCHAUD, 2008). Il a critiqué la méthodologie utilisée sur la statistique des mouvements boursiers et la modélisation des risques financiers, et notamment le modèle *Black-Scholes*. Un an plus tard, le 6 août 2009, *Nature* a publié un article intitulé «*The economy needs agent-based modelling* » écrit par FARMER et FOLEY (2009). Dans cet article, les auteurs ont écrit que face à la crise financière de 2007, les deux meilleurs modèles que le groupe économique du président des États-Unis BARACK OBAMA et ses homologues internationaux avaient utilisés, le modèle statistique empirique (modèle d'économétrie) et le modèle d'équilibre général dynamique stochastique (*DSGE*), avaient échoué face aux changements spectaculaires et à l'environnement complexe. Mais, la simulation du modèle *agents basé* pouvait, selon eux, résoudre ces problèmes. En même temps, BUCHANAN (2009) a posé une question «*Could agent-based computer models prevent another financial crisis ?*»

Étant donné que la célèbre revue *Nature* a mentionné le modèle *basé agents* à plusieurs reprises, d'autres questions ont surgi : Qu'est-ce que le modèle *basé agents* ? Pour-

1. *Nature* : une des revues scientifiques les plus anciennes et les plus réputées au monde.

quoi ce modèle économique est-il toujours mentionné dans la revue *Physique* ? Pourquoi ne trouve-t-on pas beaucoup d'articles sur le modèle *basé agents* dans d'autres grandes revues scientifiques ? Pourquoi les économistes évitent-ils de faire de la simulation en économie ?

1.2 Naissance du modèle *basé agents*

Dans les années soixante-dix du XX^e siècle, les États-Unis sont entrés dans une période où la croissance économique était plus faible qu'auparavant et l'inflation très forte, ce qui est appelé la stagflation. Pour comprendre ce phénomène et le fonctionnement de l'économie, les économistes ont suivi deux voies de recherche complètement différentes. Une fraction était composée d'économétriciens, représentée par CHRISTOPHER A. SIMS, ayant obtenu le prix Nobel d'économie en 2012. Les partisans de cette fraction ont simulé un fonctionnement macroéconomique en créant de grandes équations simultanées avec des méthodes de métrologie et de grandes quantités de données. Dans la conception du modèle, un moins grand nombre de théories économiques et d'hypothèses était considéré comme meilleur. Ces chercheurs ont préféré laisser les données parler, dire la vérité. L'autre fraction était composée d'économistes qui suivaient la pensée de ROBERT LUCAS, ayant reçu le prix Nobel d'économie en 1995. Pour ce dernier il est trop naïf de prévoir la fluence du changement d'une politique économique seulement à partir des données historiques et leurs relations. Dans un article publié en 1976, il a expliqué davantage pourquoi les suggestions de la politique basée sur le modèle économétrique à grande échelle proposées par les économistes n'étaient pas fiables. Les conclusions du modèle étaient soit insignifiantes, soit incapables de répondre aux économistes pour résoudre le problème, car les coefficients du modèle économétrique variaient à mesure des changements politiques, n'étant pas des coefficients structurels séparés. Telle était *la critique de Lucas*. À ses yeux, l'aspect dynamique manquait aux modèles économétriques.

SIMS a suivi et développé la pensée de la première fraction. L'économiste néerlandais JAN TINBERGEN, premier Prix Nobel d'économie en 1969, a tenté de construire un modèle macroéconomique afin de vérifier la théorie du cycle économique à travers des données et à l'aide de son modèle. L'économiste norvégien TRYGVE MAGNUS HAAVELMO, Prix Nobel d'économie en 1989, est un des fondateurs de la théorie de la probabilité et de l'économétrie. Il a montré que toute théorie économique devait être exprimée sous forme de probabilités, parce qu'il n'existait pas de théorie économique absolue parfaite. Mais, dans la pratique, les économistes ont rencontré des difficultés concernant la puissance de calcul, dans la mesure où il n'y avait pas d'ordinateur assez performant. Selon la théorie de KEYNES, Il existe de nombreuses incertitudes qui causent le cycle économique, il faut donc ajouter nombre de variables dans le modèle. Cependant, pour simuler le fonctionnement de la macroéconomie réelle, un modèle plus grand, plus compliqué que celui de HAAVELMO s'avérait nécessaire.

À mesure du développement de l'ordinateur individuel, la simulation en économie est devenu plus proche de la réalité. La rationalité limitée est un concept qui a été forgé par HERBERT ALEXANDER SIMON, ayant reçu le Prix Nobel d'économie en 1978. Ses travaux portaient sur l'étude du comportement d'un individu face à un choix. THOMAS JOHN SARGENT, ayant reçu le Prix Nobel d'économie en 2011, a étudié les conditions dans lesquelles les systèmes avec une rationalité limitée d'agents et d'apprenants convergeaient leurs adaptations vers des anticipations rationnelles. Ceci est considéré comme l'origine de la théorie *basé agents*. Par la suite, LEIGH TESFATSION et LEBARON ont proposé le concept de l'économie computationnelle à base d'agents. LEIGH TESFATSION a aussi créé un site d'économie computationnelle à base d'agents, regroupant les résultats des dernières études sur le modèle *basé agents* dans le monde entier.²

Néanmoins, le modèle *basé agents* a des racines qui remontent aux années quarante du XX^e siècle. C'est en effet à cette époque qu'est née la première machine automate cel-

2. <http://www2.econ.iastate.edu/tesfatsi/ace.htm>

lulaire qui simulait essentiellement les grilles des interrupteurs interagissant avec leurs voisins les plus proches (BUCHANAN, 2009). NÊME (2001) a étudié toute la pensée économique des économistes qui avaient obtenu le prix Nobel en la matière après la Seconde Guerre mondiale. Dans son livre, *La pensée économique contemporaine depuis Keynes*, les pensées sont divisées en cinq grandes catégories : les avatars du concept d'équilibre, le néo-libéralisme et le monétarisme, les hétérodoxies, l'avènement de l'économétrie et la nouvelle économie : information et progrès technique. Bien que le modèle *basé agents* n'ait pas été mentionné dans son ouvrage, nous pouvons malgré tout voir l'ombre du prototype du modèle *basé agents*. Au début des années quatre-vingt-dix, où les ordinateurs, devenus puissants, ont commencé à faire des simulations réalistes, à mesure que les experts de différents domaines y participaient, la théorie du modèle *basé agents* s'est développée, puis améliorée.

Le modèle *basé agents* était une approche nouvelle pour la recherche sur l'économie mondiale. Le principe consiste à créer un système économique par une simulation informatique : d'abord, on décrit le comportement des agents au niveau individuel en utilisant la technologie de l'intelligence artificielle, puis on construit un système économique de manière ascendante avec un grand nombre de ces individus. Le but est de résoudre des problèmes que l'économie traditionnelle ne peut elle-même résoudre, à cause du manque d'une méthode quantitative pour développer des modèles dynamiques complexes. Il s'agit d'une approche interdisciplinaire étroitement liée à l'informatique, à l'économie évolutionniste, ainsi qu'à l'économie expérimentale.

1.3 Critiques des théories économiques traditionnelles

L'économie traditionnelle étudie les relations entre les différentes activités économiques de la société humaine par *l'homo œconomicus*. *L'homo œconomicus* est considéré comme la pierre angulaire permettant d'établir et d'améliorer le système de la théorie économique

traditionnelle. Il comprend deux significations : selon la première, les gens poursuivent leurs propres intérêts, c'est-à-dire que tout le monde est égoïste ; suivant la seconde, les gens sont rationnels et taxés sur leurs actions avec une compréhension très claire. La création de ces hypothèses est attribuée à ADAM SMITH, premier grand économiste classique, connu de certains pour le dogme du pouvoir autorégulateur du marché, «la main invisible». (*Théorie des sentiments moraux 1759*) Bien que cette hypothèse ait fourni un moyen pratique sur l'analyse mathématique, elle a aussi produit de nombreux doutes, parce qu'elle a donné à l'humanité, qui est complexe, la plus haute sagesse pour chercher à maximiser les avantages. Un grand nombre de faits historiques prouvent que les individus ne sont pas des machines qui savent seulement maximiser leurs propres intérêts. SIMONE a pensé que la rationalité des gens était limitée, et il a proposé d'utiliser la rationalité limitée au lieu de complètement l'irrationalité totale. Pour l'économie traditionnelle, l'auteur a une préférence comportementale interne, complète et cohérente. PAUL ANTHONY SAMUELSON, Prix Nobel d'économie en 1970, a proposé la théorie de la préférence, révélée en 1938 dans l'optique de détacher la théorie du consommateur de tout fondement psychologique. Mais, de plus en plus d'études ont montré que la préférence de l'auteur pouvait être modifiée par l'influence de la société, le groupe de référence et les médias. Par conséquent, la base de la théorie micro-économique de l'économie traditionnelle a été largement remise en cause.

En même temps, face à la crise économique, l'analyse de l'équilibre a également présenté un défi. Pendant des années, les méthodes de recherche en économie ont toujours consisté à analyser l'équilibre. Par l'équilibre, il faut entendre que la quantité de bien fournie est égale à la quantité demandée par le rajustement des prix, le marché est alors relativement statique. Une fois que le système économique a atteint cet état idéal, une gestion plus raisonnable des ressources est formée. Mais pourquoi l'économie tendrait-elle toujours vers un équilibre stable ? Qui perturberait accidentellement les chocs exogènes ? Bien que des anticipations rationnelles correspondent à cet état, elles ne peuvent supprimer l'incertitude. Le second économiste conscient de l'incertitude est

KEYNES (JESSUA, 1991). Mais il import d'abord de mentionner FRANK KNIGHT, membre de la première école de Chicago, avec SIMONS et JACOB VINER. KNIGHT a soigneusement distingué le risque de l'incertitude. Si le premier est probabilisable et peut être couvert par l'assurance, il n'en est pas de même pour l'incertitude. Dans le monde réel, avec les fluctuations économiques, les crises économiques, les dépressions, les chocs monétaires, budgétaires, etc, il n'est pas possible de trouver un équilibre sur le long terme. L'équilibre n'est que relatif et transitoire. Ainsi, la théorie économique traditionnelle ne peut expliquer ces phénomènes macroscopiques.

Une autre critique contre la théorie économique traditionnelle est l'abus des mathématiques. La plupart de la littérature économique actuelle regorge de formules mathématiques. La plupart des prix Nobel d'économie est attribuée à des mathématiciens. La simplicité mathématique suffit-elle pour justifier l'exception en faveur de la science économique ? La Société d'économétrie a été fondée en 1930. Depuis les années soixante-dix, tous les économistes s'efforcent d'enfermer un segment de la réalité dans un petit modèle économétrique. Avec les ordinateurs personnels, l'économétrie a conquis les chercheurs individuels (toute thèse en sciences économiques doit comporter une application économétrique), les institutions financières et les organisations internationales pour réaliser leurs analyses comparatives. Cependant, le système économique est généralement dynamique et non linéaire, et parfois même, chaotique. La théorie de Keynes relève de la statique comparative, qui est un premier pas vers la dynamique (NÊME, 2001). Les premières utilisations par les économistes des modèles non linéaires remontent aux années cinquante. La théorie du chaos a connu un véritable engouement à la fin des années quatre-vingts et au début des années quatre-vingt-dix, en raison de l'évolution des faits et des idées. Bien que la science mathématique soit un outil très pertinent pour expliquer les théories économiques, elle ne peut refléter la situation réelle de l'économie. Un mouvement étudiant pour la réforme de l'enseignement de l'économie a été lancé en France en mai 2000 et s'est diffusé dans le monde entier. Les problèmes soulevés par ces étudiants s'articulaient essentiellement autour de la question de la pertinence de l'en-

seignement supérieur de l'économie pour la compréhension des problèmes économiques réels, surtout les cours économiques qui sont constitués de modèles purement mathématiques. EINSTEIN a dit que les modèles devraient être aussi simples que possible. Mais, nous devons admettre que nos modèles actuels *DSGE ad hoc* sont représentatifs des modèles macroéconomiques antérieurs. Il faut passer aux modèles qui prennent au sérieux l'interaction des agents (COLANDER, HOWITT, KIRMAN, LEIJONHUFVUD et MEHLING, 2008).

De plus, certains économistes se sont aperçus que l'économie était un système adaptatif et complexe, au même titre que la nature qui assure la survie de l'homme (HAZY, GOLDSTEIN et LICHTENSTEIN, 2007). Les interactions locales des individus microscopiques ont formé une régularité macroéconomique. Par contre, le dynamisme macroéconomique a eu un impact profond sur les individus microscopiques. La théorie du «*public choice*», élaborée par JAMES BUCHANAN, Prix Nobel en 1985, est fondée sur l'individualisme méthodologique qui ramène les interactions sociales à la somme des interactions individuelles. AMARTYA SEN, Prix Nobel en 1998, a enrichi la théorie du bien-être en reliant les objectifs sociaux aux choix individuels interdépendants (1970, *Collective choice and social welfare*; 1982 *Choice, welfare and measurement*). Bien que ce retour entre les structures microscopiques et macroscopiques soit reconnu depuis longtemps par les économistes, ceux-ci ont eu un manque de méthodes quantitatives pour faire face à ce modèle dynamique complexe. Il faudra attendre de nouveaux progrès théoriques.

1.4 Essor des sciences des systèmes complexes

Comme l'électricité à la fin du XIX^e siècle, la technologie de l'information s'est révélée l'innovation majeure des années quatre-vingts et quatre-vingt-dix. Depuis les années quatre-vingts, les grands clivages micro-macro et travail-capital ont été brouillés au profit d'une vision globale et dynamique de l'économie.

Les sciences des systèmes complexes se sont progressivement développées au cours de cette même période. Les systèmes vivants, les systèmes intelligents, les systèmes informatiques, les systèmes économiques, les systèmes sociaux, etc, sont tous des systèmes complexes. Les scientifiques ont commencé à briser les frontières entre les disciplines scientifiques, ayant effectué des recherches approfondies dans toutes les matières.

L'institut de *Santa Fe*, fondé en 1984 par plusieurs scientifiques ayant obtenu le prix Nobel, est un institut de recherche spécialisé dans l'étude des systèmes adaptatifs complexes. Depuis les années quatre-vingt-dix, ce centre, voué à l'étude des systèmes complexes adaptatifs, est connu dans le monde entier. La dynamique émergente de systèmes physiques, biologiques et sociaux est désormais étudiée, principalement au moyen de la modélisation et de la simulation *basé agents*.

La méthode principale permettant d'étudier des systèmes adaptatifs complexes à l'institut de *Santa Fe* est la simulation par ordinateur basée sur les agents. La grande différence entre la méthode traditionnelle et cette méthode est que le point de départ de l'analyse du problème de cette dernière est l'individu dans le système, mais non dans sa totalité. Nous pouvons créer un environnement économique et une société virtuelle par ordinateur en mettant en place le modèle *basé agents*. Les agents font surgir un phénomène ou un règlement à travers des interactions complexes. Par conséquent, ce phénomène macroscopique n'est pas planté par le modelleur, il est formé par le système lui-même de bas en haut. FERBER (1995) le définit comme étant « *une entité réelle ou virtuelle, évoluant dans un environnement, capable de percevoir et d'agir dessus, qui peut communiquer avec d'autres agents, qui exhibe un comportement autonome, lequel peut être vu comme la conséquence de ses connaissances, de ses interactions avec d'autres agents et les buts qu'il poursuit.* »

DE GRAUWE (2010) distingue deux types de modèle macroéconomique dynamique : le modèle *top-down* et le modèle *bottom-up*. Les modèles macroéconomiques fondés sur les anticipations rationnelles supposent des capacités cognitives extraordinaires de cha-

cun des agents. Les agents individuels utilisent de petites parties de l'information et des règles simples afin de guider leur comportement.

1.5 Recherche sur le processus d'apprentissage et évolution du comportement *basé agents*

Dans les théories économiques traditionnelles, l'analyse est basée sur les modèles mathématiques, examinant seulement les individus rationnels, avec un comportement homogène. Mais, dans les modèles *basé agents*, les individus microscopiques possèdent une capacité d'apprentissage, une capacité inductive, une capacité d'adaptation et une rationalité limitée (ARTHUR, 1994).

Les trois conditions posées par ARROW sont clairement résumées par SEN (1970), à savoir :

1. l'université, la fonction doit exister dans tous les cas de préférences individuelles, sans forcément dégager l'unanimité ;
2. la cohérence parétienne : si quelqu'un préfère x à y, la fonction sociale doit aussi préférer x à y ;
3. l'indépendance : si dans deux alternatives, les préférences individuelles concernant x et y sont les mêmes, elles doivent aussi être les mêmes dans la fonction sociale.

Celles-ci peuvent être considérées comme une partie de la composition du prototype du modèle *basé agents*.

Comment créer un mécanisme d'apprentissage pour les agents selon la pensée humaine ? Les agents peuvent utiliser leur propre expérience ainsi que celle de leurs voisins pour guider leur prise de décision. Dans «*Learning from Neighbours*», BALA et GOYAL

(2000) ont développé un *framework* général pour étudier la relation entre la structure de ses voisins et le processus d'apprentissage social. HERLEMONT (2004) a modélisé le comportement des agents sous la forme de graphes du modèle de CONT-BOUCHAUD, la plate-forme *basée agents* est développée en JAVA.

Actuellement, le mécanisme d'apprentissage de l'agent est généralement établi par des algorithmes génétiques et des réseaux de neurones qui appartiennent à l'intelligence artificielle. HOLLAND (1992) a montré que les comportements des interactions entre les agents étaient semblables à ceux des commerçants réels. À *Wall Street*, la majorité des managers et des institutions prennent les décisions en fonction de l'expérience du passé lorsqu'ils doivent effectuer des transactions face aux bénéfices ou aux risques potentiels. Et il leur faut apprendre les expériences des autres. La capacité d'adaptation est définie comme l'individu qui peut apprendre ou accumuler des expériences pendant une interaction avec l'environnement et d'autres individus, et peut changer leur propre structure ou leur propre comportement en fonction des leçons apprises. TESFATSION (2006) a étudié différents problèmes économiques en utilisant la différence entre les comportements des agents dans le système économique, tout en combinant leur capacité d'apprentissage afin de simuler et d'analyser.

1.5.1 Brève histoire du développement et de la recherche du modèle *basé agents*

LANE (1993) a décrit une classe de modèles, appelée monde artificiel (*AWs artificial worlds*), lesquels sont conçus pour donner un aperçu d'un processus nommé l'organisation hiérarchique émergente (*EHO*). Dans son article, il présente les idées de *EHO* et *AWs*, et traite certains des problèmes interférentiels impliqués en essayant de se renseigner sur *EHO* par la construction et l'étude des propriétés de *AWs*. Il a d'abord proposé le concept du modèle économique calculable à base d'agents. Selon lui, on peut établir un système

économique du marché de la simulation en utilisant un modèle économique calculable à base d'agents que les individus peuvent utiliser pour des expériences répétitives.

À partir de ce moment-là, le modèle *basé agents* a principalement été utilisé pour l'analyse des évolutions de différents marchés et de l'impact de son régime associé. Il a été étudié dans des domaines différents, tels que les marchés automatiques, le commerce et la gestion, la politique économique, les marchés de l'électricité, l'économie financière, les marchés du travail, l'économie politique, l'organisation industrielle, entre autres.

Le secteur de l'électricité au niveau mondial est actuellement en cours de modifications substantielles. Ces changements tentent d'éloigner le secteur de l'électricité de la tarification en fonction des coûts de gestion et d'entrer dans un marché concurrentiel, car les prix peuvent refléter pleinement la relation entre l'offre et la demande dans un marché très réglementé. Le but de ces changements est de fournir de meilleures incitations pour les participants du secteur, afin de contrôler les coûts et d'introduire l'innovation. En été 2000, l'effondrement du marché de gros de l'électricité de la Californie aux États-Unis a montré que, sans la mise en œuvre préalable de tests suffisants, personne ne peut prévoir les orientations du marché. Ainsi, nombre de chercheurs en énergie ont souligné qu'il fallait utiliser de nouveaux outils de modélisation computationnelle à base d'agents pour construire un système restructuré de l'électricité à modéliser, lorsque les réseaux commerciaux des négociants en interaction stratégique et les organismes de réglementation apprennent à fonctionner dans le temps sur les réseaux de transport réaliste.

AMIN (2002) a montré comment les réseaux de communication et de coopération des agents logiciels et intelligents pouvaient être utilisés pour gérer et s'adapter à des systèmes distribués complexes, et comment les collections d'agents pouvaient être employées pour acheter et vendre l'électricité, et participer au marché électronique. Il a introduit un modèle et la simulation «*Electric Enterprise*» développés par EPRI (*Electric Power Research Institute*) en utilisant des individus autonomes, des agents adaptatifs pour représenter les composantes industrielles possibles et les personnes morales, proprié-

taires de ces composantes. ZHOU, CHAN et CHOW (2007) ont étudié quatre packages de simulation *basés agents* pour les marchés de l'électricité. Ainsi, *SEPIA (Simulator for Electric Power Industry Agents)* est considéré comme le premier modèle *basé agents* pour le marché de l'électricité et *EMCAS (Electricity Market Complex Adaptive Systems)* est vu comme étant le plus puissant. *STEMS-RT (Short-Term Electricity Market Simulator - Real Time)* et *NEMSIM (National Electricity Market Simulation System)* ont également été examinés, mais leurs principales caractéristiques et leur limites ont été discutées.

L'environnement du marché du travail est dynamique, stochastique, inaccessible, non-déterministe, non-markovien et continu. LEWKOVICZ et KANT (2008) ont utilisé le modèle économique proposé par CAHUC et CARCILLO afin de construire un modèle *basé agents* du travail en France. Trois types de contrat de travail appartiennent à leur modèle : le *CDD*, le *CDI* et le *CNE*. Ils ont également introduit quatre types d'agents, l'agent "personne", l'agent "compagnie", l'agent "correspondant" et l'agent "gouvernement". À la fin, ils ont trouvé que l'objectif de ces agents était d'améliorer leur situation et de choisir un état avec l'utilité espérée supérieure. Le résultat a montré qu'avant l'introduction du contrat *CNE*, le taux de chômage s'était stabilisé et juste après, ce taux diminuait et une seconde stabilisation intervenait à un niveau supérieur. On pouvait toutefois trouver ces mêmes résultats dans le modèle économique où les économistes expliquaient que le taux de chômage s'était stabilisé à un niveau supérieur sur le long terme.

Le modèle *basé agents* a été largement utilisé en écologie, car l'économie écologique comprend des interactions entre le comportement humain et l'environnement. Il apparaît comme un système socio-écologique, y compris la dynamique de la population, le comportement du groupe et sa spécialisation, la gestion forestière et maritime, la planification de la conservation, etc. BITHELL et BRASINGTON (2009) ont analysé la sylviculture et la culture dans les communautés agricoles de subsistance. Dans leur modèle, la croissance des cultures, les précipitations, le cycle hydrologique impliquent des évaluations. La forêt est représentée comme un modèle basé sur l'individu avec différentes espèces

d'arbres. La communauté des ménages est représentée comme un modèle *basé agents* qui fonctionne sur un paysage hydrologique spatial. SCHLÜTER et PAHL-WOSTL (2007) ont examiné la résilience des systèmes de culture et de pêche irrigués sous différentes modalités de gouvernance. Ils ont souligné que des scénarios climatiques dans les différentes politiques d'utilisation de l'eau avaient été explorés et qu'une communauté d'agents avait pris des décisions qui affectaient l'écologie aquatique et de la pêche.

LIU *et al.* (2007) ont synthétisé les études sur la démonstration des méthodes utilisées et les résultats obtenus dans la complexité des systèmes humains et naturels couplés sur les cinq continents, en Afrique, en Chine, aux États-Unis, dans une zone près du Brésil et en Suède. Ils ont traité explicitement les interactions complexes et la rétroaction entre les systèmes humains et naturels. En particulier, ils ont mesuré non seulement des variables écologiques et des variables humaines (caractéristiques du paysage, habitat de la faune et de la biodiversité, processus socio-économiques, réseaux sociaux, agents et constructions de la gouvernance multi-niveaux), mais aussi des variables qui visaient les composantes naturelles et humaines (collecte du bois et utilisation des services écosystémiques).

Le modèle *basé agents* est également appliqué à la politique économique. Il y a quarante ans, l'économiste THOMAS SCHELLING (1971) a élaboré une théorie qui explique la persistance de la ségrégation raciale dans un environnement avec de plus en plus de tolérance raciale. Ce fameux modèle a commencé avec quelques pièces de monnaie sur une grille de huit fois huit et quelques hypothèses très simples concernant les préférences individuelles. SCHELLING a utilisé les *micromotives* pour expliquer les macrophénomènes, mais, n'ayant pas d'outil pratique pour étudier la ségrégation, il n'a pas énormément développé sa recherche pendant 30 ans. Cependant, avec le modèle *basé agents*, il est beaucoup plus facile d'analyser le comportement global des individus dans les modèles où eux-ci sont engagés dans la même transaction. SANDER, SCHREIBER et DOHERTY (2000) ont développé le concept de SCHELLING. Ils ont examiné les fonctions des préférences dérivées à base de données empiriques sur la composition raciale des

voisins et ont ajouté une variété de facteurs putatifs dans leur choix de logement. Ces chercheurs ont également montré comment tester des modèles *basé agents* avec une date empirique. Selon eux, un modèle *basé agents* pourrait donner un aperçu des modèles empiriques de la ségrégation et de la déségrégation à la fin du XXI^e siècle. La loi de puissance décrit une relation mathématique entre deux quantités. CEDERMAN (2003) a proposé un modèle *basé agents* de la guerre et de l'état-formation présentant des régularités selon la loi de puissance. Le résultat prouve que le comportement sans échelle dépend d'un processus de changement technologique qui conduit à des décisions stochastiques et dépendantes contextuelles de la guerre.

1.5.2 Premières recherches sur le modèle *basé agents* en finance

Le marché financier est un marché couramment utilisé pour les applications et les recherches du modèle *basé agents*. Le modèle traditionnel du marché financier reposant sur les anticipations rationnelles et la théorie des marchés efficients est très commode sous forme mathématique, néanmoins, il n'existe pas de modèle qui puisse expliquer entièrement les différents phénomènes au sein des marchés financiers réels, par exemple, le phénomène de queue lourde dans la distribution des rendements ou encore la corrélation entre le volume des transactions et la volatilité. Par rapport aux autres modèles traditionnels, dans les marchés de la simulation à base du modèle *basé agents*, les chercheurs peuvent comprendre tous les détails de l'activité du marché. Objectivement, on possède des conditions de base pour une analyse approfondie de ces questions.

Le premier modèle *basé agents* à avoir eu une grande influence sur le marché financier est le modèle du marché boursier artificiel, réalisé par ARTHUR, HOLLAND, LEBARON, PALMER et TAYLER (1996), différent des modèles traditionnels de trading. Ces chercheurs ont renoncé à l'hypothèse selon laquelle chaque agent doit avoir toutes les informations et une rationalité parfaite. Ils ont imaginé que ces derniers pouvaient apprendre d'affi-

lée des informations historiques, et ensuite modifier leurs prévisions pour le cours des actions. Ce marché boursier artificiel est un système en constant changement et jamais équilibré. La relation entre les agents est une synergie de la concurrence et de la coopération. Ce modèle a simulé avec succès la réalité psychologie du marché boursier, ainsi que des phénomènes de mutation non linéaire sur les points de ces marchés.

La fondatrice du Site Web du modèle *basé agents* (2000), Leigh Tesfatsion (2005), a d'abord proposé le concept de l'économie de calcul à base d'agents, avant de le définir comme suit :

« *Agent-based computational economics (ACE) is the computational modeling of economic processes (including whole economies) as open-ended dynamic systems of interacting agents* (TESFATSION, 2006).»

Ce site regroupe les études les plus récentes des économistes du monde entier sur le modèle *basé agents* et il fait figure d'autorité actuellement.

De nombreux articles sur les caractéristiques comportementales des participants du marché et leurs effets sur les propriétés des marchés sont apparus ces dernières années. OMURTAG, KNIGHT et SIROVICH (2000) ont introduit un modèle *basé agents* afin de modéliser une population de participants, dont l'équation était utilisée pour étudier le comportement des interactions mimétiques, dérivant rigoureusement du modèle avec un agent unique sous-jacent sur le marché financier. Ils ont utilisé la théorie des champs de décision élaborés par des chercheurs en psychologie quantitative (DIEDERICH 1997) et étudié uniquement des taux de l'ordre d'achat ou de vente lorsque des fonctions de l'information avaient été saisies. Leurs résultats montrent que le taux d'ordre qui vient directement de la simulation fluctuait autour du taux de l'ordre donné par l'équation de la population tout au long des simulations à l'équilibre pendant des durées transitoires, avec des systèmes stables ou instables.

BOOKSTABER (2012) a analysé les menaces de la stabilité financière avec le modèle

basé agents. Il a critiqué le modèle de VaR (*Value-at-Risk*) et celui des tests de stress. Le premier était largement appliqué dans le marché financier avant la grande crise de 2007 aux États-Unis afin de mesurer le risque de perte d'un portefeuille sur le temps horizon donné et la probabilité. Le second a été critiqué parce qu'il n'avait pas pu anticiper les chocs extrêmes produits pendant la crise et qu'il avait ignoré la dynamique entre les secteurs qui avaient été touchés, en fin de compte. BOOKSTABER (2012) a établi une application financière du modèle *basé agents* où il a inclus des agents comme des sociétés de gestion d'actifs qui investissent dans les banques et les produits dérivés, et qui les guident pour financer leurs investissements. Dans son modèle, il a aussi intégré des agents clés, des règles de comportement pour les agents, des leviers politiques, des chocs et une certaine vulnérabilité, ainsi que des données pour définir les conditions initiales et des applications politiques. Pour lui, le modèle peut être utile à l'analyse de la stabilité financière, car les agents peuvent représenter les entités actuelles dans le système financier.

1.6 Le modèle *basé agents* et la physique

Dans un épisode d'une sitcom américaine *The Big Bang Theory*, a été racontée une plaisanterie que la plupart des spectateurs n'ont pas comprise : un paysan a demandé à un physicien de résoudre un problème avec ses poules dans sa ferme, car elles ne pondaient plus. Après avoir fait des calculs, le physicien a annoncé qu'il avait trouvé une solution, mais que cette solution était seulement valable pour des poules sphériques dans un *vacuum*. Cette petite histoire signifie que les calculs des scientifiques ne s'appliquent pas à la vie réelle ; les physiciens utilisent un modèle très simple pour simuler le monde réel. Cependant, tout calcul, dans les recherches scientifiques, sert le modèle que les scientifiques choisissent, plutôt que le monde réel. Le modèle très simplifié est assez correct. Par exemple, si nous devons calculer l'orbite céleste, il nous faut simplifier toutes les étoiles et

les planètes, et les considérer comme des particules sans volume. Parfois, il est nécessaire de simplifier le modèle. Par exemple, si nous voulons simuler le climat mondial, nous ne devons tenir compte que des courants océaniques et de l'impact du mouvement des glaciers arctique et antarctique. Ainsi, faut-il inclure les changements des nuages dans le modèle ? L'impact des plantes distribuées ? Celui de l'éruption volcanique islandaise ? On ne saurait penser à tout, en raison de la puissance de calcul limitée. On ne prend pas un modèle parce que le monde est complexe, mais parce que notre observation du monde est subjective. Les théories économiques traditionnelles ont toujours été fondées sur un jugement subjectif ; de ce fait, elles sont souvent inefficaces, ce qui n'est pas surprenant. En Chine, dans le livre *Tao Te Ching*³, LAO TSEU dit :

«*Le Tao, qui peut être dit, n'est pas le vrai Tao ; Le nom, qui peut être nommé, n'est pas le vrai nom.*»

Dans *Entretiens de Confucius*⁴, chaque fois, des gens différents demandent à Confucius : «qu'est-ce que la bienveillance ?». Et il en donne une définition correspondant aux différentes personnes.

En fait, cela indique le mode de travail de la recherche sur la science, il ne faut jamais poursuivre «la vraie réalité», mais seulement la simuler avec différents modèles, de manière continue. Ainsi, quand un scientifique dit qu'il se passe quelque chose dans le monde réel, il indique que les variables correspondant à cette chose sont arrivées à une situation dans son modèle. Et le modèle est un choix naturel pour étudier l'évolution et la simulation.

Parmi tous les modèles scientifiques, la théorie en physique est la plus réussie. Par exemple, l'électrodynamique quantique n'est pas le dernier modèle dans le monde de la physique, elle considère simplement toutes les particules comme des balles. Bien

3. *Tao Te Ching* : un ouvrage classique chinois écrit par Lao Tseu autour de 600 av. J.-C, le sage fondateur du taoïsme

4. *Entretiens de Confucius* : une compilation des discours de Confucius (551-479 av. J.-C.) et de ses disciples ainsi que des discussions entre eux.

qu'elle ne tienne compte ni des interactions à l'intérieur du noyau, ni de la gravité, c'est un modèle parfait. Avec quelques équations très simples, il est possible de décrire quasiment tous les phénomènes, hormis la gravité et l'extérieur du noyau ; et ce modèle est très précis. RICHARD FEYNMAN⁵ a écrit dans un livre la valeur de calcul du moment magnétique de l'électrodynamique quantique par la théorie : $1,001\ 159\ 65 \pm 0,000\ 000\ 000\ 2$ magnéton de Bohr, mais la valeur mesurée expérimentalement est $1,001\ 159\ 65 \pm 0,000\ 000\ 000\ 03$ magnéton de Bohr. Cette erreur est équivalente si l'on calcule la distance de Paris à Pékin, à travers les hémisphères d'Est en Ouest, la différence entre le résultat calculé et le résultat réel est infiniment petite. Par conséquent, nous pouvons prévoir très précisément chaque éclipse solaire, nous pouvons intercepter les missiles, nous pouvons même déposer un détecteur à la surface de Mars en utilisant la méthode de contrôle à distance. Ces modèles "physiques" imparfaits donnent des résultats qui, eux, sont parfaits. Certaines personnes pensent à tort que la science devrait fournir des réponses précises, mais le fait est que la majorité des modèles concernant d'importants problèmes ne peut y parvenir.

La crise financière de 2008 montre que les économistes ne l'avaient pas prévue. Cependant, nous ne devrions pas être trop critiques à l'égard des économistes uniquement parce qu'ils ne sont pas arrivés à prévoir les tendances macro-économiques. En fait, la crise économique, aux yeux du public, contient différents types de cause et d'effet. En d'autres termes, aucun modèle économique n'a la capacité de comprendre toutes les causes qui peuvent être quantifiées et influencer une crise. Celui qui a prévu avec succès a par hasard inclus la cause de la crise dans son modèle. Mais cela ne ressemble pas à la théorie de la relativité, car il peut ne plus fonctionner la fois suivante. Cependant, nous ne pouvons nier le rôle des économistes sur ce point. En fait, selon la perspective de l'histoire économique, la fréquence d'apparition des crises a été réduite grâce aux progrès continus des connaissances et des améliorations théoriques de l'économie. On n'exige pas que l'économiste fasse des prévisions précises sur les tendances économiques comme prévoir

5. Un des physiciens les plus influents de la seconde moitié du XX^e siècle

les pointes de descendre du détecteur sur Mars. Mais les économistes peuvent indiquer aux gens, grâce à la recherche, comment prendre des décisions optimales sur la base des informations et des connaissances limitées. Le travail de la plupart d'entre eux consiste à trouver une solution optimale dans un ensemble limité en mathématique. Et ceci est précisément ce que peut faire le modèle *basé agents*.

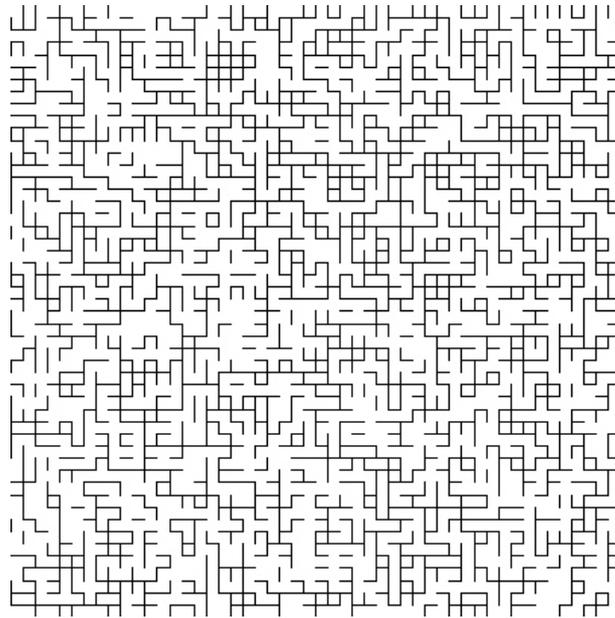
Percolation

Au cours des dernières décennies, la théorie de la percolation a été largement appliquée en physique, en sciences des matériaux, en réseaux complexes, ainsi que dans d'autres domaines. Cette théorie a été introduite par S.BROADBENT et J. HAMMERSLEY en 1957 comme un modèle d'écoulement de fluide à travers un milieu poreux désordonné.⁶ Imaginons que l'on verse un verre d'eau au sommet d'un matériau poreux. S'il existe suffisamment de petits canaux pour faire passer l'eau, alors on peut trouver un chemin qui guide l'eau à couler du sommet vers le bas. Bien que le modèle ait rapporté une quantité de preuves heuristiques et expérimentales importantes dans de nombreux phénomènes, Il est toujours plus difficile d'analyser avec rigueur que de prévoir. Mais, en 1980, après 20 ans de recherche sur cette théorie, H.KESTEN a prouvé le premier résultat que la probabilité critique de percolation de liaison sur Z^2 est $1/2$.

En économie, la théorie de la percolation fournit une approche complémentaire pour étudier les systèmes complexes. Elle permet d'examiner la propagation d'une information entre les agents dans un réseau, et notamment dans le modèle *basé agents*, dont on sait que les interactions entre les agents sont très importantes. On fait une simulation d'après les résultats des interactions. Mais comment les agents échangent-ils leurs informations ? Habituellement, pour les chercheurs du modèle de la diffusion de l'information, il existe deux modèles principaux : le modèle mathématique au niveau macro-économique et le

6. voir la figure 1.1

FIGURE 1.1 : Détail d'une percolation de liaison sur le réseau carré en deux dimensions, la probabilité de percolation est de 51%. Source : Percolation theory, Wikipedia.



modèle de simulation au niveau micro-économique. Les chercheurs ont analysé les facteurs qui influent sur la diffusion de l'information, la vitesse de propagation, le degré de propagation, entre autre, sous différents angles. Le premier modèle mathématique au niveau macro-économique a été créé par BASS (1969). Il analyse les règles de certaines variables macro-économiques à mesure du changement de temps et les relations entre les différentes variables macro-économiques. Ce genre de modèle nécessite de beaucoup de données adéquates, mais la difficulté de collecter des données ainsi que leur volatilité causent souvent des problèmes comme l'efficacité et l'opportunité de l'évaluation. De plus, le modèle de BASS ne différencie pas les agents, les considérant tous en tant qu'un ensemble. Il obtient alors une relation de changement pour l'ensemble. Bien qu'il puisse avoir un résultat de la propagation de l'ensemble, il ne peut explorer le processus spécifique ni le mode de propagation. En revanche, chaque fois que l'on ajoute une variable, le modèle devient compliqué et plus difficile à résoudre. Par conséquent, on doit le simplifier à l'aide de nombreuses hypothèses, par exemple, en ignorant l'hétérogénéité entre les agents, en supposant que tous les individus sont homogènes, en considérant

que chacun est soumis aux mêmes effets internes et externes, ou encore en excluant les relations sociales des agents, etc. Tout cela est toutefois incompatible avec la réalité. Il existe des limites. Le modèle de simulation au niveau micro-économique à l'aide du modèle de percolation tient compte, avec plus de détails, des différences de préférence des agents concernant les réseaux et les informations en réalité. Il considère aussi d'autres facteurs telle que l'hétérogénéité des agents qui influencent la vitesse et l'étendue de la diffusion de l'information. Il permet également de surmonter les inconvénients des hypothèses du marché homogène et révèle le mécanisme de la propagation de l'information au niveau micro-économique. Le modèle *basé agents* peut créer une structure de réseau micro-social en utilisant le modèle de percolation. À travers ce réseau, le modèle *basé agents* peut construire une forte influence sur les agents par leurs préférences et les actions des voisins avec diverses fonctions poids (voisin direct et voisin indirect). Dans notre étude, les agents vont construire leur réseau par une règle de connexion spécifique. Dans le Chapitre 2 de cette thèse, les décisions adoptées par les agents sont affectées non seulement par des facteurs externes, mais aussi par leurs propres interactions des agents. Chaque décision que prend un agent est affectée par l'ensemble de ses voisins, et chaque décision du voisin est aussi affectée par l'ensemble des décisions de ses propres voisins. Tous les agents de la société peuvent être considérés comme une seule cellule dans la grille à deux dimensions ci-après. Dans la figure 1.2, 1 et 0 représentent les décisions de l'agent après la propagation de l'information. 1 indique que l'agent prend la décision oui et 0 représente non. L'idée principale du modèle *basé agents* est que chaque agent décide du comportement qu'il adaptera au moment suivant selon la situation actuelle externe et interne. La figure 1.2 est le résultat d'un réseau de 100 agents d'après 1 000 000 de simulations, avec une probabilité de 50% de connexion du voisinage.

Du point de vue du temps continu, dans un intervalle du temporel très court t , la probabilité qu'un agent change d'idée est la formule suivante :

$$f_i(t)\Delta P_t = 1 - (1 - C_t)(1 - Vt)^{n_i(t)} \quad (1.1)$$

FIGURE 1.2 : Distribution du réseau des agents, $P=0.5$, $L=10$, $S=1,000,000$

1	1	0	1	1	1	0	1	0	0
1	0	0	0	0	0	0	0	0	1
0	1	0	1	1	0	0	0	0	1
0	0	0	0	0	0	1	0	1	1
1	0	1	0	0	0	0	0	1	0
0	1	1	1	0	1	0	1	1	0
0	0	0	1	1	0	1	1	1	1
1	0	1	0	0	0	0	0	1	0
0	0	0	0	0	0	1	1	0	0
0	0	0	0	1	1	0	0	0	0

$f_i(t)$ est appelé influence du marché sur l'agent i au moment t . On peut ainsi comprendre l'équation : durant le moment ΔP_t , la probabilité de l'adoption d'une information pour l'agent i par un choc extérieur est $C_{\Delta P_t}$, alors, la probabilité de ne pas adopter cette information est $(1 - C_{\Delta P_t})$; la probabilité de l'adoption d'une information pour l'agent i par ses voisins est $V_{\Delta P_t}$, alors, la probabilité de ne pas adopter cette information est $1 - V_{\Delta P_t}$; $n_i(t)$ est le nombre de ses voisins. Étant donné que le choc extérieur et l'influence des comportements des voisins internes sont indépendants, selon les règles de calcul des probabilités, pour les événements indépendants, la probabilité totale est égale à la multiplication des événements indépendants. Ainsi la probabilité du refus d'une information pour l'agent i est $(1 - C_{\Delta P_t})(1 - V_{\Delta P_t})^{n_i(t)}$. Donc, la probabilité de l'adoption d'une information pour l'agent i est $1 - (1 - C_{\Delta P_t})(1 - V_{\Delta P_t})^{n_i(t)}$.

Puisque ΔP_t est infiniment petit, l'équation (1.1) peut être réécrite comme suit :

$$f_i(t)\Delta P_t \approx 1 - (1 - C_{\Delta P_t})[1 - Vn_i(t)\Delta P_t] \quad (1.2)$$

$$= 1 - [1 - C_{\Delta P_t} - Vn_i(t)\Delta P_t + CVn_i(t)(\Delta P_t)^2] \quad (1.3)$$

$$= C_{\Delta P_t} + Vn_i(t)\Delta P_t - CVn_i(t)(\Delta P_t)^2 \quad (1.4)$$

Hormis une petite quantité d'ordre élevé, nous obtenons :

$$f_i(t)\Delta P_t \approx C_{\Delta P_t} + Vn_i(t)\Delta P_t \quad (1.5)$$

Dans la figure 1.2, le réseau comprend 100 agents, $10 * 10$, au moment t , supposant qu'il y ait $N(t)$ agents qui adoptent l'information, alors, pour les $100-N$ agents qui la refusent, il y aura $pN(t)$ agents qui adopteront l'information :

$$pN(t) = \sum_{i=1}^{100-N} [f_i(t)\Delta P_t] \quad (1.6)$$

$$= \sum_{i=1}^{100-N} [C\Delta P_t + Vn_i(t)\Delta P_t] \quad (1.7)$$

$$= [C(100 - N) + V \sum_{i=1}^{100-N} n_i(t)]\Delta P_t \quad (1.8)$$

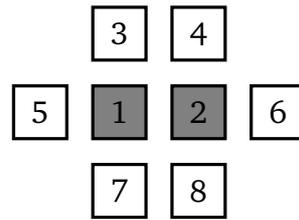
Si le nombre de voisins d'une cellule est k (soit $k = 4$, Voisinage de von Neumann, soit $k = 8$, Voisinage de Moore), alors :

$$\sum_{i=1}^{100-N} n_i(t) = kN(t) - 100k\left(\frac{N(t)}{100}\right)^2 \quad (1.9)$$

L'équation (1.5) calcule l'ensemble des voisins qui adoptent l'information pour un

agent qui la refuse. Cela indique que l'on calcule le nombre 1 autour d'un agent représenté par 0. Mais ce nombre n'est pas égale à $N(t)$, car pour un agent connu, s'il y a plus d'un voisin qui refuse l'information, quand on calcule, cet agent est calculé plusieurs fois, et le nombre maximum de répétitions est k . Mais pour le montant de $kN(t)$, il nous faut enlever la statistique des voisins qui adoptent l'information pour un agent qui l'adopte soit le deuxième élément à droite de l'équation. La figure 1.3 explique cette équation.

FIGURE 1.3 : *Les agents adoptent l'information*



Dans la figure 1.3, 1 et 2 représentent les agents qui adoptent l'information, 3, 4, 5, 6, 7 et 8 représentent ceux qui la refusent. L'agent 1 a été calculé 4 fois lors de sa connexion avec les agents 2, 3, 5 et 7. Parmi les 4 fois, les connexions avec les agents 3, 5, 7 sont valables, celle avec l'agent 2 doit être retirée. Il en est de même pour l'agent 2. Ainsi le nombre enlevé est $2 * \mu(t)$, $\mu(t)$ étant le nombre des deux agents voisins ayant adopté l'information dans le réseau. Pour un réseau avec 100 agents, si l'on tient compte des limites périodiques, le nombre total de connexion est $100k/2$. Chaque connexion correspond à deux cellules, la probabilité pour que chacun d'elle soit un agent qui adopte l'information est $N(t)/100$, alors :

$$2 * \mu(t) = 100M * \frac{N(t)}{100} * \frac{N(t)}{100} \quad (1.10)$$

On fait la substitution de l'équation (1.6) dans l'équation (1.5) :

$$\frac{pN(t)}{\Delta P(t)} = C(100 - N(t)) + 100V \frac{N(t)}{100} (100 - N(t)) \quad (1.11)$$

TANAKA (2002) a construit un réseau des interactions des traders avec le modèle de percolation dans son modèle *basé agents*, afin de reproduire la distribution à queue épaisse. Il a défini et calculé le prix plancher et le prix plafond comme des valeurs critiques pour les prix des actions de *NIKKEI 225* dans une procédure de percolation en 2002. Il a trouvé que les informations privées n'avaient pas été révélées à tous les agents, mais seulement à une partie d'entre eux sur le marché. Le comportement des traders dépend des décisions de leurs voisins dans une interaction locale, qui est éloignée du théorème de la limite centrale supposant que les agents ont une vue illimitée et une certaine rationalité. COLLARD, COLLARD et STATNER (2012) ont analysé les mécanismes de flux d'informations à travers la procédure de la percolation. Ils ont présenté un modèle de mobilité synthétique et créé un monde artificiel avec des agents hétérogènes dans un environnement *basé agents* programmable. La percolation à travers un réseau peut trouver des seuils qui assurent la connexion et la propagation de l'information. Dans le modèle *ER*, un modèle de mobilité du modèle *basé agents*, les agents explorent les zones de leur espace selon les règles individuelles, montrant qu'il avait reproduit des modèles du monde réel. FRENKEN (2006) a aussi proposé le modèle de percolation pour analyser la structure complexe de l'interaction, entre des agents dans une invention collective en évitant la «sur-paramétrisation». Le modèle de ZEPPINI *et al.* (2013) comprend deux facteurs importants de la diffusion économique : la diffusion de l'information et l'adoption des décisions. Ce modèle de percolation montre une transition de phase d'une diffusion à un régime de non-diffusion avec un prix seuil qui dépend principalement de la structure du réseau et de la distribution des prix des réservations.

Chapitre 2

Les méthodes et les procédures générales de la modélisation du modèle *basé agents*

Le modèle *basé agents* est la modélisation informatique des processus économiques (y compris les économies entières) en tant qu'un système dynamique ouvert avec des agents interagissants. Il s'agit d'une approche ascendante (dite *bottom-up*). Bien que ce procédé de modélisation soit plus scientifique, il apporte une certaine quantité de difficultés pour les modélistes, car il n'existe pas de méthode générale qui puisse identifier les règles microscopiques correspondantes, en principe. En d'autres termes, afin d'obtenir une sorte de propriété émergente, les modélistes ne sont pas parvenu à trouver une règle microscopique avec une méthode de raisonnement logique, ne pouvant qu'utiliser à plusieurs reprises des méthodes par l'observation répétée, l'induction, l'hypothèse et les tests pour vérifier leurs idées. Par conséquent, en résumé, nous pouvons voir les étapes et les méthodes générales de la modélisation du modèle *basé agents*.

Dans le processus de la modélisation du modèle *basé agents*, habituellement, on uti-

lise les étapes suivantes : l'induction, l'observation, la modélisation informatique, le test par ordinateur, ainsi que l'explication des étapes à restaurer. D'abord, il importe de saisir la direction et les points importants sur les priorités de recherche par induction, observation et abstraction sur le système économique réel. Ensuite, les modélistes doivent résumer certaines hypothèses plus simples et établir un modèle informatique visant ces hypothèses selon le système économique réel. Puis, il s'agit de faire toutes sortes de recherches sur la simulation informatique par ordinateur, et d'observer le comportement émergent du système économique de ce modèle. Les chercheurs comparent aussi le comportement émergent dans le modèle avec les phénomènes du système économique réel pour expliquer la loi du système économique actuel. On peut également modifier les paramètres et les règles du modèle afin d'examiner les différentes lois afin de connaître l'effet du système économique virtuel porteur de l'environnement macro-économique représenté par les paramètres. Enfin, les règles générales sont résumées, y compris les recommandations politiques visant le système économique réel.

Dans l'ensemble, dans des circonstances dites normales, la modélisation du modèle basé agents comprend les étapes suivantes :

(1) Observer le système économique réel et abstraire une série d'hypothèses (A) ;

(2) Établir le modèle de l'agent individuel, ainsi que la règle de l'interaction (R) entre agents d'après Les hypothèses (A) ;

(3) Observer le résultat de l'émergence et le résultat macroscopique dynamique (O) du système économique artificiel par la simulation de l'ordinateur ;

(4) Selon les observations (O), obtenir une conclusion afin d'expliquer le phénomène de l'économie réelle ;

(5) Modifier les paramètres et les règles du modèle, observer comment les paramètres affectent le résultat de l'émergence du modèle.

Parmi ces étapes, la plus importante est la première, les hypothèses prélevées du monde réel, car elle détermine l'utilité du modèle *basé agents*. Toutefois, une certaine littérature de la modélisation de ce modèle analyse purement les caractères du système économique à partir du modèle lui-même, sans se soucier du fait que ce système économique artificiel puisse bien refléter l'économie réelle. Bien que l'étude de ce genre de modèle ait été très critique, les modélistes ont souligné que la valeur du modèle *basé agents* consistait à étudier diverses possibilités économiques, et non nécessairement l'économie réelle. Parfois, l'étape 5 joue également un rôle clé, car l'on atteint l'objectif des tests du modèle *basé agents* grâce à l'ajustement en permanence des paramètres du modèle. Le système *basé agents* lui-même fournit une plate-forme de tests pour les scientifiques, à tel point que certaines personnes disent qu'il s'agit d'un laboratoire économique. Si une décision capable d'influencer le système économique a été testée à plusieurs reprises, et a fait l'objet d'une démonstration complète dans le laboratoire *basé agents*, ce modèle apportera moins de risques pour la prise de décision réelle.

2.1 Les technologies clés de la modélisation du modèle *basé agents*

Une question centrale dans la recherche du modèle *basé agents* est l'analyse de la modélisation visant un phénomène économique particulier ou le système économique. Bien que ce modèle soit essentiellement un système multi-agents, son objectif est de simuler des phénomènes économiques au lieu de résoudre des problèmes dans le système de l'intelligence artificielle. Ainsi, la modélisation du modèle *basé agents* nécessite de méthodes uniques et de technologies clés. Dans ce chapitre, nous allons nous concentrer sur trois aspects : l'établissement du modèle d'agent individuel, l'interaction entre agents et l'analyse du résultat de l'émergence du modèle *basé agents*.

2.1.1 Le modèle d'agent

Un agent est une entité caractérisée par le fait qu'elle est, au moins partiellement, autonome. Ce peut être un processus, un robot, un être humain, etc. Bien qu'il ne soit pas entièrement défini, les individus admettent généralement que l'agent est un programme d'ordinateur disposant d'une autonomie, de compétences sociales, de réactivité et de spontanéité. Ainsi, un système expert complexe avec des capacités cognitives peut être considéré comme un agent, un programme court avec des réponses souples peut être conféré à titre d'agent. Essentiellement, le modèle *basé agents* est un système multi-agents. Mais il considère l'agent comme un outil afin d'imiter le comportement humain par rapport au système de l'intelligence artificielle avec le modèle *basé agents*, préférant utiliser la concurrence et la coopération entre des agents pour résoudre un problème de calcul dans la vie. Ainsi, l'agent, dans le modèle *basé agents*, est différent de celui utilisé par le système de l'intelligence artificielle. L'agent, dans le modèle *basé agents*, est nommé agent économique afin de le distinguer.

Le modèle de l'agent économique, dans le modèle *basé agents*, peut habituellement présenter un corps mobile dans un système économique. Il peut être une personne, une entreprise ou une famille. Il peut produire différentes catégories d'agents selon divers prélèvements de ses activités. Si l'on divise les modèles des agents économiques selon la fonction exécutive, le niveau du comportement et celui de la complexité, on peut les diviser en trois catégories : les agents réactifs, les agents réfléchis et les agents adaptatifs.

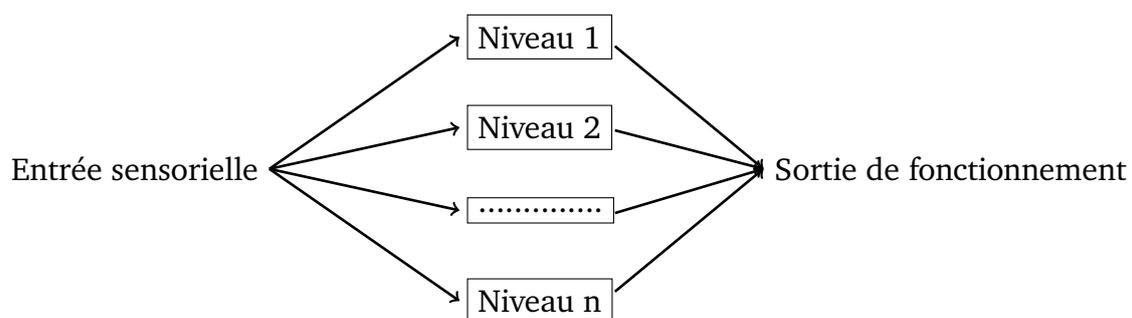
Les agents réactifs Ce genre de modèle d'agent suit généralement une règle très simple, par exemple, *<if> ... <then> ...* Mais lorsque un grand nombre d'agents suivant cette règle interagissent, certains phénomènes qui rassemblent socialement les humains émergent. En effet, un *home œconomicus* du modèle néo-classique en économie équivaut à un agent réactif. Car dans un environnement donné, l'agent n'accomplit sa décision que pour pour-

suivre la maximisation de son utilité.

Bien qu'ils soient tous des agents réactifs, le modèle crée divers agents économiques d'après différentes règles. La structure de règles la plus courante est la structure plate, ce qui signifie que toutes les règles *<if> ... <then> ...* sont au même niveau, et chaque fois, l'on ne peut activer qu'une seule règle. Bien évidemment, il existe d'autres agents réactifs avec différentes structures, par exemple, l'architecture subsumption et l'automate fini qui contient un état interne.

BROOKS (1986) a introduit une architecture de subsumption en tant qu'architecture robotique réactive fortement associée à la robotique basée sur le comportement. Nous pouvons diviser les règles des agents en plusieurs niveaux, chacun ayant une réaction différente. La règle du comportement au niveau le plus bas a une priorité plus élevée. Ainsi, l'agent peut choisir et peser le pour et le contre et en même temps, plusieurs règles du comportement sont appliquées. En conséquence, on considère que l'agent avec une architecture de subsumption possède une structure multi-couches de traitement parallèle à l'information. Regardons le schéma ci-dessous :

FIGURE 2.1 : *Agent avec une architecture subsumption*



Même si cette méthode semble très simple, elle est très efficace. Elle a donc été adoptée par de nombreux modèles *basé agents*. Le modèle *Sugarscape* est un type d'architecture de subsumption (EPSTEIN et AXTELL, 1996).

L'automate fini (*Finite State Machine*, appelé *FSM*) rejoint l'état interne dans des

règles alors qu'il élargit la complexité du comportement des agents (LEWIS et PAPADIMITRIOU, 1997). En général, une règle réactive a la structure suivante :

$$if < input > \quad then < action > \quad (2.1)$$

Ici, $<input>$ représente l'information d'entrée du capteur détectée par l'agent, $<action>$ est la sortie sélectionnée effectuée par l'agent. Dans le modèle *FSM*, si l'on définit une agrégation S représentant l'ensemble des états internes d'un agent, et en supposant que la règle réactive non seulement relie les informations d'entrée, mais aussi leur état interne, cette règle peut être ainsi écrite :

$$if < input > \quad and \quad < currentstate > \quad then \quad < action > < nextstate > \quad (2.2)$$

$<current state>$ représente l'état interne actuel de l'agent et $<next state>$ l'état interne dans l'instant suivant, si l'agent adopte cette règle. Nous constatons clairement que la règle réactive transformée par le *FSM* est non seulement déterminée par la réaction et les informations d'entrées du monde extérieur, mais qu'elle est aussi reliée par son état interne. Ainsi, l'agent a une mémoire limitée. Évidemment, plus nombreux sont les états internes d'un agent, plus complexe est son comportement.

FOGEL (1991) a utilisé le *FSM* pour guider l'agent afin de tester de manière répétitive le jeu du dilemme du prisonnier, en vue de trouver une stratégie d'optimisation efficace. GMYTRASIEWICZ et LISETTI (2002) ont utilisé l'état interne de l'agent pour représenter ses différents états émotionnels, de sorte à créer un agent avec une émotion simple.

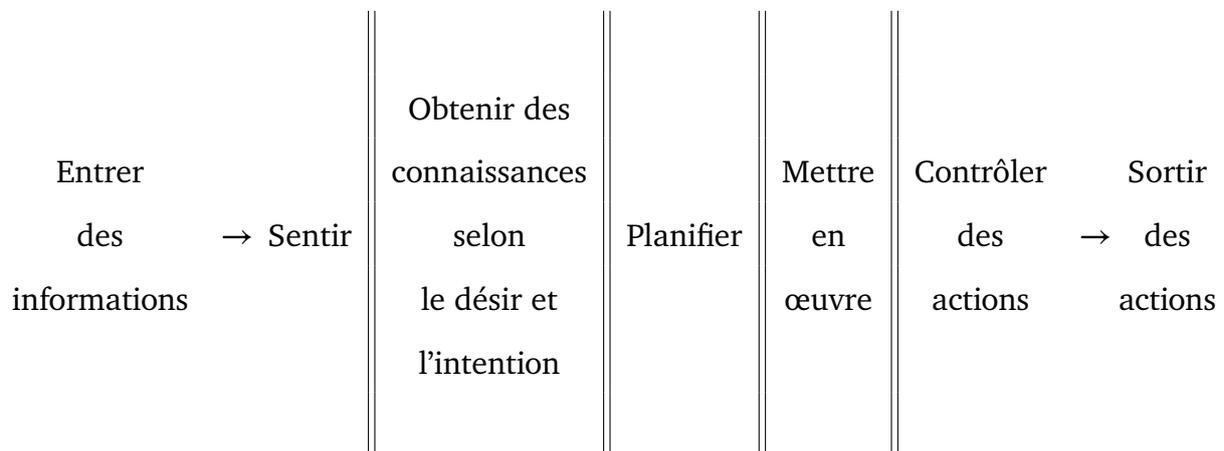
Dans des applications réelles de la modélisation du modèle *basé agents*, tous les agents réactifs suivant la règle $<if> \dots <then> \dots$ sont pré-conçus par le modéliste. Bien que cela réduise l'adaptabilité de l'agent, car la méthode est simple, directe et efficace, cela est plus pratique pour une simulation d'agents à grande échelle. De ce fait, la majorité

des modélistes du modèle *basé agents* optent pour cette méthode.

Les agents réfléchis En général, les agents réfléchis ont une structure de raisonnement plus rigoureuse. Ainsi, chaque étape de l'action de l'agent réfléchi est le résultat d'une analyse rationnelle et d'une pensée attentive. La théorie du raisonnement pratique *BDI (Belief, Desire, Intention)* est utilisée comme base du modèle des agents réfléchis. Ce modèle comprend les connaissances, les objectifs, les désirs et d'autres définitions préconçues (HANNEBAUER, 2002). Toutes les actions des agents réfléchis comprennent les étapes suivantes : observer les informations depuis l'extérieur, coder ces informations en tant que leurs propres symboles reconnaissables internes, obtenir un programme de l'action en l'analysant avec leurs connaissances et leurs objectifs, et enfin, la mise en œuvre de ces actions.

En résumé, les agents réfléchis possèdent une structure de transmission des informations en série par rapport aux agents réactifs. La figure 2.2 montre cette structure :

FIGURE 2.2 : *Structure du modèle de l'agent réfléchi*



Certains modèles d'agents réfléchis sont destinés à la simulation du processus cognitif humain réel. Ils combinent la capacité d'apprendre et de s'adapter avec un processus réfléchi complexe. Les agents réfléchis ont donc des caractéristiques et des comportements anthropomorphiques, comme la connaissance, le raisonnement, la planification

ou l'apprentissage, (NEWELL, 1994). Ces modèles sont toujours utilisés pour simuler une organisation multi-agents, où les différents agents réfléchis peuvent accomplir des tâches complexes individuellement. Dans ce système d'organisation, chacun joue un rôle spécial et possède des connaissances et des expériences dans un domaine précis.

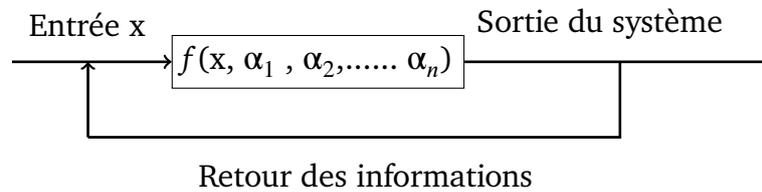
En somme, la caractéristique du modèle des agents réfléchis est de se rapprocher du processus individuel de pensée et de cognition humaine. Mais, ce modèle consomme beaucoup de ressources informatiques et est difficile à mettre en œuvre. Par conséquent, ces modèles sont toujours utilisés pour simuler les petites organisations. Les agents utilisés dans le modèle d'équipe de conception virtuelle (*Virtual Design Team*, appelé VDT) (KUNZ, CHRISTIANSEN, COHEN, JIN et LEVITT, 1998) et dans le modèle *Orgahead* (LOUIE, CARLEY, HAGSHENASS, KUNZ, LEVITT *et al.*, 2003) sont mentionnés ci-dessus. Le modèle *Soar* (LAIRD, NEWELL et ROSENBLOOM, 1987) et le modèle *Plus Soar* (CARLEY, KJAER-HANSEN, NEWELL et PRIETULA, 1992) appartiennent à ce type de système comprenant des agents réfléchis.

Les agents adaptatifs Dans ce type de modèle, la capacité d'adaptation et l'auto-apprentissage de l'agent sont les clés. L'agent peut s'adapter constamment à l'environnement virtuel où il vit et changer sa structure selon le résultat de sa cognition. Les agents adaptatifs sont divisés en deux types : l'agent avec une structure simple, l'agent avec une structure complexe.

Le modèle de l'agent adaptatif simple peut être considéré comme un agent réactif avec des paramètres réglables. D'abord, il réagit selon les informations d'entrée depuis l'extérieur, puis il sort des informations. Ensuite, il règle les paramètres internes α_1 , $\alpha_2, \dots, \alpha_n$ selon la rétroaction de l'opération de sortie, afin de modifier les règles d'action $f(\alpha_n)$. Ce processus d'ajustement des paramètres selon la rétroaction est l'apprentissage de l'agent. Dans ce modèle, le réglage des paramètres est simple et direct, mais leur flexibilité et leur adaptabilité sont faibles ; de plus, il y a une limite au niveau de l'autonomie

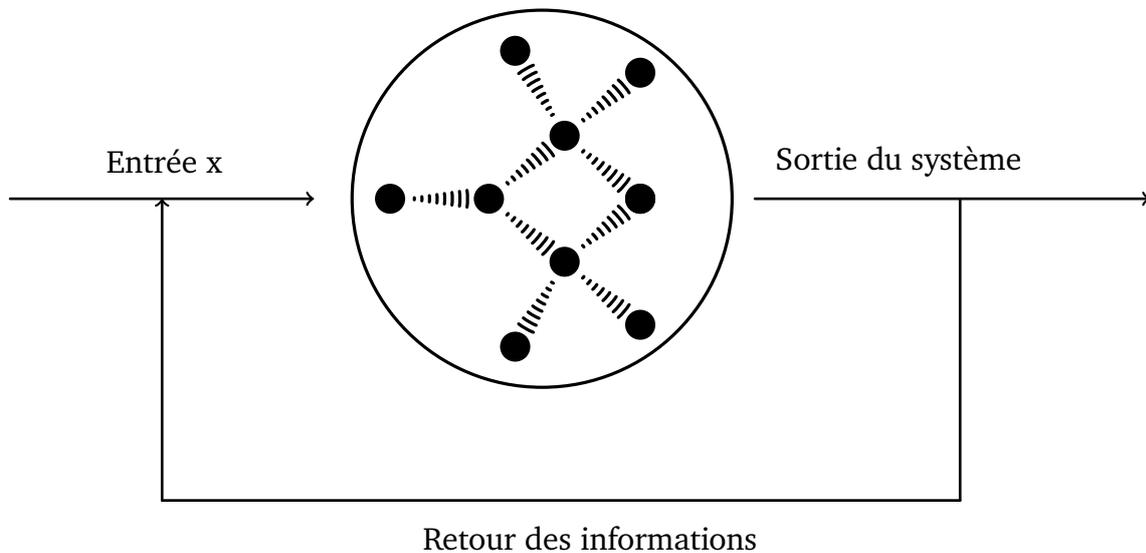
et de l'intelligence de l'agent, comme le montre la figure 2.3 :

FIGURE 2.3 : Agent adaptatif simple



La structure complexe d'un autre type de modèle d'agents adaptatifs consiste à considérer l'agent lui-même comme un système complexe, dans lequel des éléments interagissent pour effectuer des calculs compliqués, puis sortent des informations. L'agent peut entrer à nouveau des informations de rétroaction dans ce système afin d'en réguler la structure interne. De cette manière, le processus d'apprentissage, dans le changement de la structure d'adaptation, est différent, comme l'indique la figure 2.4 :

FIGURE 2.4 : Agent adaptatif complexe



Actuellement, le système de classification génétique, la programmation génétique, le réseau de neurone BP (*Backpropagation*, appelé *Rétropropagation du Gradient*) etc. sont

utilisés comme des méthodes pour établir des modèles économiques d'agents adaptatifs. Le système de classification génétique peut être considéré comme un système très complexe, composé de multiples classificateurs concurrents. Le fonctionnement des algorithmes génétiques assure la capacité d'innovation du système et réalise des exigences du système d'apprentissage. La programmation génétique utilise quelques fonctions simples, comme des blocs de construction. Leurs combinaisons incessantes constituent un programme informatique, où l'innovation et l'apprentissage se font par des algorithmes génétiques. Le réseau de neurones *BP* utilise un grand nombre de neurones interactifs afin de former un système de réseaux complexes dans lequel l'agent finit son processus d'étude par l'algorithme de rétropropagation. TEMA (2000) a construit un modèle d'agent adaptatif avec un réseau de neurone *BP* artificiel afin de prévoir la tendance des prix (LUNA et STEFANSSON, 2012). ARTHUR, HOLLAND, LEBARON, PALMER et TAYLER (1996) ont utilisé le système de classification génétique pour créer un petit système complexe permettant d'établir le modèle interne de l'agent. EDMONDS et MOSS (1997) ont réalisé un modèle d'agent par la programmation génétique.

2.1.2 Les interactions entre agents

Le système économique réel est composé d'individus, leur communication mutuelle et leurs interactions constituant une économie complexe. De même, dans le système économique virtuel artificiel, l'interaction entre les agents peut constituer une dynamique économique virtuelle complexe. Ainsi, l'interaction entre agents deviendra inévitablement la clé dans la modélisation du modèle *basé agents*.

Le mode interactif Actuellement, on divise l'interaction entre agents en trois types selon le mode interactif : la communication directe, l'interaction directe et l'interaction indirecte.

A) La communication directe Cette communication permet à deux agents d'échanger leurs informations directement, pouvant affecter l'état interne et le comportement de chacun. Pour l'instant, l'étude sur l'origine des langues avec le modèle *basé agents* utilise cette approche (CANGELOSI et PARISI, 2012). On a comparé les signaux abstraits transmis entre deux agents aux informations contenues dans le monde virtuel. Ces informations se réunissent pour former un concept ou produire une culture entre agents différents. Le biologiste et éthologiste britannique, *Clinton Richard Dawkins*, a inventé le mot « mèmes », (équivalent culturel du « gène »), afin de décrire comment les principes darwiniens pouvaient être étendus pour expliquer la façon dont les idées et les phénomènes culturels se répandent. La communication entre agents peut être considérée comme un processus consistant à copier, à distribuer et à varier constamment un « mème ». Dans l'étude du modèle *basé agents*, les gens peuvent directement pratiquer la modélisation sur un « mème ». Ils utilisent une longue chaîne de caractères 0 et 1 pour représenter un certain type de « mèmes », puis étudier la règle de la propagation des « mèmes » entre agents (GABORA, 1997).

B) L'interaction directe Dans l'interaction directe entre agents, l'action d'un agent peut d'abord changer l'état d'un autre agent, puis provoquer le changement de comportement d'un troisième agent. Un agent en influence sur un autre par sa propre action ou son propre choix dans l'interaction. Comparativement, dans la communication, un agent touche directement d'autres agents dans une communication abstraite formée par un instrument à corde à l'intérieur du mental. C'est la plus grande différence entre l'interaction et la communication. Autrement dit, la communication est une communication de « cœur à cœur » et l'interaction est une interaction de « corps à corps ».

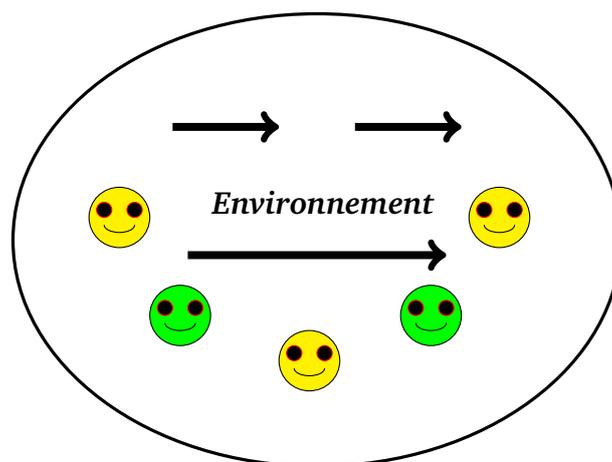
Comment expliquer que des agents terminent leur interaction par des actions différentes ? La réponse consiste à définir différentes règles du jeu. En d'autres termes, on réunit deux ou plusieurs agents pour effectuer le jeu d'une structure fixe. À la différence

de la théorie du jeu traditionnel, le modèle *basé agents* ne prête pas attention à la solution d'équilibre dans le jeu, le considérant comme une règle, et les différents agents doivent interagir selon les règles.

C) L'interaction indirecte On parle d'interaction indirecte lorsqu'il n'y a ni communication ni interaction réelle entre les agents, mais ceux-ci peuvent faire interagir indirectement à l'aide de l'environnement comme véhicule de la communication. Il existe deux cas *basé agents* dans l'environnement pour les agents.

L'un est l'environnement local. Dans ce cas, un agent ne peut interagir qu'avec l'environnement à côté de lui, mais plusieurs peuvent faire une interaction indirecte par le véhicule de l'environnement intermédiaire. La simulation de fourmis à la recherche de nourriture est un des exemples les plus notables (DROGOUL, 1993). Les fourmis ne sont pas organisées hiérarchiquement et ne disposent d'aucun moyen de communication globale. Mais comment les fourmis trouvent-elles de la nourriture ? Elles communiquent avec leurs congénères en utilisant des odeurs chimiques comme les phéromones. Leur comportement est entièrement régi par le hasard et par leur environnement immédiat. La figure 2.5 montre cette relation entre les agents et l'environnement.

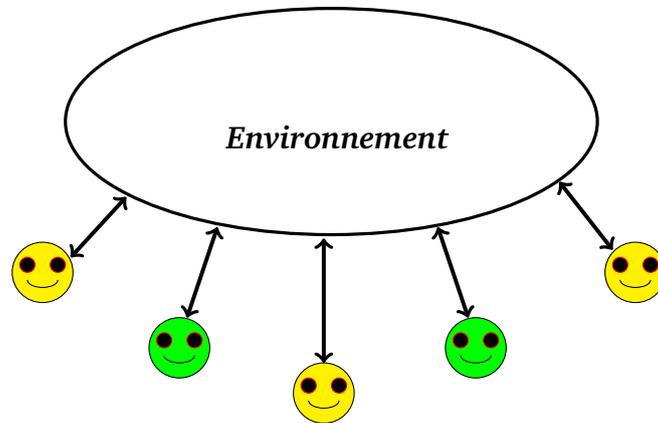
FIGURE 2.5 : *Interaction entre les agents et l'environnement (1)*



L'autre cas est l'environnement macroéconomique mondial. Cela signifie que l'en-

semble du comportement de chaque agent détermine une certaine propriété de l'environnement macroéconomique mondial, et vice versa, cet environnement macroéconomique mondial peut affecter chaque agent. La figure 2.6 exprime cette relation d'interaction.

FIGURE 2.6 : *Interaction entre les agents et l'environnement (2)*



Le modèle du marché financier artificiel (*Santa Fe artificial stock market*, appelé *SF-ASM*) étudié par LEBARON (2001) appartient à cette relation interactive. Dans ce modèle, le comportement dynamique de chaque boursicoteur détermine la tendance globale du prix des actions. D'autre part, le prix des actions détermine la prévision et la prise de décision de chaque agent. Ainsi, un tel système composé par ces agents constitue un système fermé et co-évolutif.

Topologie Que ce soit la communication directe, l'interaction directe ou l'interaction indirecte, les agents doivent achever l'interaction dans un ordre logique ; il s'agit de la structure topologique. On prend toujours deux types de structure topologie dans le modèle *basé agents*.

A) L'automate cellulaire Tous les agents sont disposés dans un monde de la grille unidimensionnelle ou bidimensionnelle. Seules les deux agents voisins dans le monde de la grille peuvent faire une interaction.

FIGURE 2.7 : *Automate cellulaire unidimensionnel*

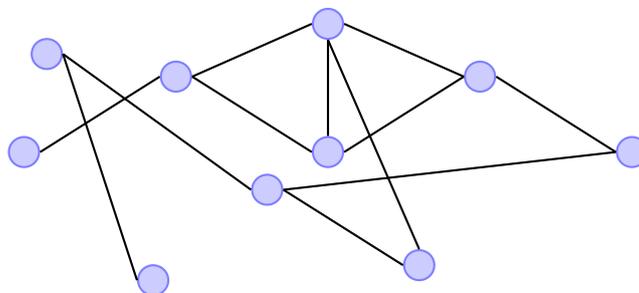


La figure 2.7 présente un automate cellulaire unidimensionnel (WOLFRAM, 1984), dans laquelle le carré noir est là où se trouve l'agent actuel. Celui-ci ne peut qu'interagir avec les deux agents voisins dans les carrés gris. Le modèle d'une émergence du rôle politique étudié par AXELROD et HAMILTON (1981) est un automate cellulaire unidimensionnel.

Mais, dans la simulation du modèle *basé agents*, l'automate cellulaire bidimensionnel est fréquemment utilisé en tant que structure topologique, laquelle peut facilement introduire la simulation dynamique du système sur un écran d'ordinateur. Le modèle *Sugarscape* est un modèle typique d'interaction, utilisant un automate cellulaire bidimensionnel.

B) Le modèle de réseau Dans ce modèle, l'espace des activités des agents est un réseau complexe (WATTS, 1999). Normalement, chaque agent est distribué sur le sommet d'un réseau, et seuls les sommets reliés par des fils sont considérés comme des voisins. C'est-à-dire que seules les deux nœuds de réseau interconnectés par des fils peuvent interagir, comme l'illustre la figure 2.8.

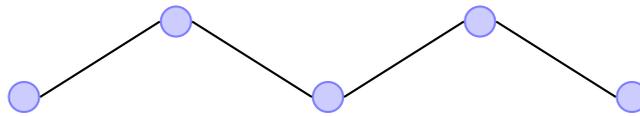
FIGURE 2.8 : *Un réseau complexe*



La structure topologique du réseau possède un très haut degré d'abstraction. En fait,

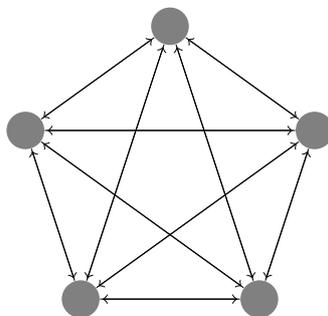
la structure topologique de l'automate cellulaire mentionné ci-dessus est la structure d'un réseau spécial. Par exemple, l'automate cellulaire unidimensionnel équivaut à la structure du réseau dans la figure 2.9, dans laquelle, chaque nœud est un carré de l'automate cellulaire, des lignes représentant les deux cases adjacentes. Évidemment, n'importe quelle structure de l'espace avec certaines dimensions est équivalente à une structure du réseau (WOLFRAM, 2002).

FIGURE 2.9 : *Une structure du réseau équivalant à un automate cellulaire unidimensionnel*



Une seconde structure de réseau est le réseau maillé (ou le maillage en réseau), comme présenté à la figure 2.10. Il s'agit d'un agent qui peut interagir avec tous les autres agents. Le test du jeu de dilemme du prisonnier est effectué dans ce réseau maillé (AXELROD, 1997). La structure interactive de l'automate cellulaire représente une caractéristique d'un effet local. Cela signifie que certains renseignements nécessitent d'une interaction entre plusieurs agents qui réalisent la transmission. En outre, chaque agent dispose d'un nombre fixe de voisins dans ce type de modèle. Mais, dans le réseau maillé, toutes les informations peuvent être transmises directement d'un agent à un autre. Cela indique que tous les agents sont tous contigus. Cependant, la majorité de la structure de réseau réside entre la structure topologique et la structure du réseau maillé.

FIGURE 2.10 : *Un réseau maillé*



Le modèle de WATTS (1999) pour le phénomène petit-monde est une structure de

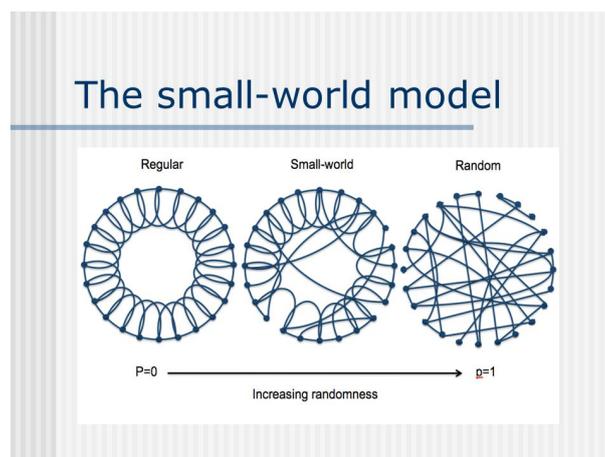
réseau. Il possède une structure topologique ressemblant à celle d'un réseau de relations humaines, de réseaux d'informations, de réseaux internet, etc, dans le monde réel. Plus particulièrement, leurs propriétés statistiques sont très similaires. Un réseau de petit-monde présente deux caractéristiques :

- la distance moyenne entre toute paire de sommets est faible ;
- le niveau de clustering local est élevé, c'est-à-dire que les sommets sont généralement très connectés à leurs voisins immédiats.

Par définition, il est nécessaire que le graphe de départ soit connexe : un chemin doit exister entre toutes les paires de sommets. La figure 2.11 illustre cette procédure pour un graphe en forme d'anneau, dans lequel chaque nœud est initialement connecté à ses k voisins de l'anneau les plus proches.

Le réseau de petit monde se situe entre le réseau complètement aléatoire et le réseau complètement régulier. La recherche sur l'interaction des agents dans un réseau de petit monde a une signification pratique, car le réseau de petit monde est très similaire au réseau de relations humaines et au réseau d'informations.

FIGURE 2.11 : Le réseau du petit monde



2.2 L'analyse des résultats de l'émergence du modèle *basé agents*

L'objectif principal de la modélisation du modèle *basé agents* est de comprendre les règles générales du système économique de la réalité à travers le modèle informatique. Bien que le phénomène d'émergence du modèle *basé agents* puisse bien expliquer le monde réel, la poursuite de l'émergence n'est pas notre but ultime. En effet, nous devons analyser et comprendre ces résultats d'émergence à l'aide d'une variété de techniques. Ainsi, la clé dans le modèle *basé agents* est de savoir comment analyser et classer les résultats d'émergence obtenus.

À la différence des modèles économiques classiques, la caractéristique principale du modèle *basé agents* est d'utiliser la complexité du système économique générée par ordinateur pour traiter la complexité du système économique dans la réalité. En d'autres termes, comme le modèle *basé agents* lui-même est un modèle très complexe, nous devons développer des méthodes analytiques et sophistiquées pour analyser la complexité du modèle lui-même. Actuellement, les gens utilisent fréquemment la méthode de l'analyse des règles individuelles, celle de l'analyse statistique des séries temporelles et la méthode de l'observation et de l'analyse d'émergence de l'animation graphique. Ces trois méthodes principales et d'autres, plus spéciales, permettant d'étudier le résultat d'émergence du modèle *basé agents*.

2.2.1 Analyse des règles individuelles

Si l'on prend l'agent adaptatif pour réaliser la modélisation du modèle *basé agents*, les règles du comportement de l'agent peuvent parfois changer à mesure que l'agent s'adapte et apprend. Car l'individu doit ajuster ses propres règles de comportement pour s'adapter à divers environnements, instants. Ainsi, le changement de comportement de l'agent

reflète la tendance du changement du système.

Nous pouvons définir la règle d'adaptation et le comportement individuel à analyser afin de prévoir une tendance évolutive de l'ensemble du système. Par exemple, si un agent utilise le système de classification pour faire la modélisation, on peut imprimer toutes les règles de classification des agents pour comprendre quel comportement ils adoptent dans des circonstances particulières.

Cependant, ce procédé n'est pas toujours efficace. D'une part, une certaine approche de la modélisation de l'agent ne comporte pas de règles explicites. Par exemple, les réseaux de neurones artificiels ne montrent pas ces comportements en cas d'isolement de l'environnement extérieur. D'autre part, il y a beaucoup d'individus dans le système, et chacun procède à un changement adaptatif dans chaque cycle de simulation. Il est donc difficile de prélever les règles à tout moment pour chaque agent.

2.2.2 Analyse statistique des séries temporelles

Le modèle *basé agents* simule un système économique, de sorte que ce dernier est en passe de connaître une histoire totalement unique. En représentant cette histoire par des données, le modèle *basé agents* peut sortir une pluralité de séries chronologiques. Nous pouvons analyser une série chronologique générée par le modèle *basé agents* et suivre cette méthode pour analyser la série chronologique du système économique réel. Tout d'abord, nous pouvons prendre un indicateur afin d'examiner le modèle *basé agents*, comme par exemple, le *GDP*, le prix d'un certain produit, etc, puis nous sortons cet indicateur en tant que donnée dans chaque cycle de simulation. Enfin, le fonctionnement du modèle à long terme génère une série chronologique. Nous pouvons connaître les tendances du développement et des états périodiques du modèle *basé agents* dans cette série chronologique.

La méthode des séries temporelles peut également être utilisée pour analyser non seulement le système global du modèle *basé agents*, mais aussi la tendance d'un certain état ou d'un indicateur d'un individu au fil du temps.

2.2.3 Analyse de l'animation graphique

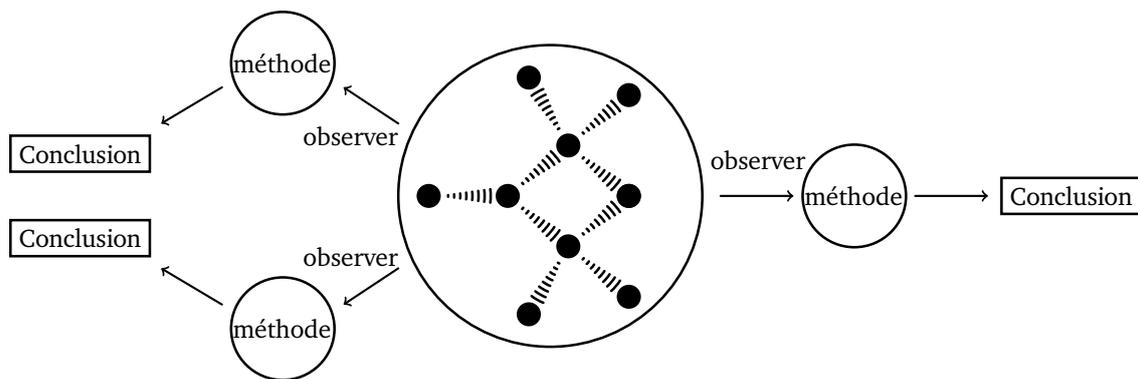
Dans le modèle *basé agents*, il existe nombre de propriétés et de règles qui sont difficiles à décrire et à traduire en formule. Cependant, ces propriétés et ces règles sont uniquement des caractéristiques qui représentent des avantages du modèle *basé agents* par rapport au modèle classique. Même si l'on ne peut pas décrire ces propriétés et ces règles, elles peuvent être observées par l'animation graphique. Cependant, cette méthode comporte une limite : elle ne peut qu'être utilisée dans le système du modèle *basé agents* qui possède une interface d'affichage graphique. Pour un grand nombre de systèmes qui n'ont pas d'interface d'affichage graphique, nous pouvons faire des animations graphiques afin de montrer la relation entre des variables et la règle de changement, cachées dans le modèle *basé agents* à l'aide de moyens artificiels. Avec la caractéristique de l'image visuelle de la méthode d'animation graphique, nous pouvons montrer et exprimer des connaissances qualitatives dans le système économique complexe.

2.2.4 Autres méthodes

Étant donné que le modèle *basé agents* est un objectif complexe et énorme, il contient beaucoup de connaissances et de règles que l'on ne peut observer d'un coup. Il est donc très nécessaire de développer une variété de méthodes pour analyser les résultats d'émergence. Nous pouvons dire que toutes les méthodes analytiques largement utilisées dans le système économique réel peuvent l'être dans l'analyse du système du modèle *basé agents*. Par exemple, dans le modèle *Sugarscape*, l'auteur a utilisé le coefficient de *Gini*,

la courbe de *Lorenz*, l'analyse dynamique du réseau, l'analyse des courbes de l'offre et de la demande, etc. Une grande quantité de méthode de recherche en économie ont prouvé leur efficacité. On peut présenter la relation entre le modèle *basé agents* et les différentes méthodes d'analyse par la figure ci-dessous :

FIGURE 2.12 : Utilisation du modèle basé agent



En résumé, le modèle *basé agents* est un système complexe, il est comme une géométrie présente dans un espace multidimensionnel. Dans l'observation et l'analyse de chaque approche, c'est comme si l'on projetait cette géométrie de l'espace multi-dimensionnelle dans un espace de dimension de base où l'on peut analyser et observer. Ainsi, il existe de nombreuses méthodes de cette projection, chacune ne pouvant refléter que quelques-unes des propriétés du modèle. Mais avec cette projection, nous pouvons comprendre davantage le modèle *basé agents*.

Cette section montre que le système économique est un système adaptatif complexe, susceptible d'être étudié par simulation à base des agents. Telle est la base théorique du modèle *basé agents*. Ensuite, visant la technologie clé de la modélisation du modèle *basé agents*, cette section propose une discussion spécifique et divise ces technologies clés en trois parties : la modélisation du modèle *basé agents*, l'interaction des agents du modèle *basé agents* et l'analyse du résultat d'émergence.

2.3 Critiques du modèle *basé agents*

Il existe aussi des critiques pour le modèle *basé agents*. Dans son article de 2015, « Les simulations multi-agents, une nouvelle approche du social ? », PABLO JENSEN indique que le modèle *basé agents* ne s'oppose pas aux théories économiques traditionnelles, mais il les rend complexes. Dans un modèle *basé agents*, les agents sont comme de petits robots qui suivent des règles simples, n'étant pas contrôlés par le niveau collectif. L'auteur considère que le modèle *basé agents* est très simple et utile, tel un microscope théorique afin d'explorer des modèles implicites. Cette approche explique la société réelle en complexifiant les modèles. Les chercheurs expliquent que tel est le cas pour toutes les tentatives d'étude de la société, et JENSEN s'est interrogé sur l'intérêt d'expliquer des relations sociales !

De notre point de vue, la société humaine est elle-même un système complexe. Le modèle *basé agents* ne complique pas les théories économiques traditionnelles basées sur les mathématiques. L'objectif de la création d'un modèle est de simplifier l'analyse de systèmes complexes. Si un modèle est complexe pour reproduire tous les détails du système réel, il n'est pas plus facile à comprendre que le système réel. En Europe, il existe des simulations parallèles, restaurant parfaitement les phénomènes complexes. Mais, le travail de conception du modèle de simulation est très lourd. Il est donc nécessaire d'avoir un système d'exploitation spécial à grande échelle. Les chercheurs ordinaires ne disposent pas de cet outil pour faire des expériences. Néanmoins, nous pensons que cette pratique n'est pas nécessairement bonne. Si l'on peut saisir quelques parties de lois clés, elle devrait être en mesure de refléter la plupart des problèmes. Après avoir achevé le modèle, nous devons le simplifier, en réduisant les petits détails inutiles, en distinguant les variables clés des variables sans importance. Pour les questions concernant la complexité du modèle *basé agents*, nous proposons un modèle simple qui capture les lois clés, répétant le motif et la variation que nous étudierons dans les chapitre 3 et 4. La clé est le mécanisme

de la propagation de l'information durant les interactions des agents.

À l'heure actuelle, la modélisation du modèle *basé agents* est utilisée dans divers domaines, comme l'économie, la finance, la robotique, la chaîne d'approvisionnement, etc. Cependant, même dans les meilleures revues d'économie, peu d'articles sur ce sujet sont publiés. La majorité des articles publiés sur ce point était lorsque Madame TESFASION était chef adjoint du Journal «*Journal of Economic Dynamic and Control*», pendant deux ans. Après cette période, les articles économiques sont devenus peu nombreux. En dehors de l'économie, les articles traitant ce sujet sont nombreux. Le modèle *basé agents* est un ajustement naturel pour étudier le problème de l'évolution. Ainsi, les revues de biologie publient beaucoup de ces articles, mais l'économie évolutionnaire est en effet très minoritaire en économie. On peut trouver des documents sur le modèle *basé agents*, mais la plupart ne décrivent qu'une application spécifique ou qu'un résultat de ce modèle n'expliquant pas suffisamment comment les chercheurs ont réalisé la procédure durant la modélisation du modèle *basé agents*. C'est un problème pour ceux qui apprennent ce modèle en économie. Mais, si le modèle *basé agents* est assez simple, on peut utiliser des plate-formes *basée agents* comme *NetLogo*, *Repast* et *AnyLogic* pour résoudre les difficultés. Dans un livre récent, *Agent-Based Modelling in Economie*, écrit par HAMILL et GILBERT (2015), les auteurs expliquent comment utiliser les trois plate-formes. Si nous voulons faire une simulation avec nos propres idées et nos propres règles, il nous faut bien maîtriser l'informatique et l'économie. Pour la modélisation du modèle *basé agents* en utilisant la programmation informatique, la capacité de mettre en œuvre un modèle mathématique est exigée, le calcul numérique et la programmation parallèle doivent aussi être maîtrisés. De plus, il est également nécessaire de comprendre l'optimisation des processus et la maintenance du fonctionnement de l'ordinateur, parfois, il faut aussi maîtriser le fonctionnement de la plate-forme informatique. Ces compétences obligatoires sont un défi énorme pour les économistes. Pour les jeunes chercheurs, il faut du temps et de l'argent pour apprendre. Toutefois, le modèle *basé agents* est un domaine émergent, et il n'est pas raisonnable de prévoir d'obtenir des résultats significatifs dans un court

délai. Quant aux économistes expérimentés, bien qu'ils possèdent cette capacité, ils ne prennent pas garde au problème de méthodologie, se concentrant plutôt sur les problèmes politiques qu'ils doivent éviter. (JOSEPH EUGENE STIGLITZ, prix Nobel en 2001) Car, une fois que cette approche a montré des résultats explosifs, elle est susceptible de renverser des idées et des théories validées au cours des années précédentes. Dans le Chapitre 3, nous expliqueront la programmation d'un modèle *basé agents* en C++ et des optimisations.

Une autre critique pour le modèle *basé agents* est qu'il montre le fonctionnement des agents dans un état non-équilibré et instable. AXELROD (2006) a déjà noté ce problème :

«ABM can be a hard sell. Since most formal theorists equate models with mathematical models, it is not surprising that some of them are hard to convince about the appropriateness and value of an agent-based simulation.»

Le principe du modèle *basé agents* est de définir comment les agents peuvent réagir. Ainsi, nous supposons implicitement un équilibre dans ce modèle, car si de nouvelles réactions sont possibles après les interactions entre les agents, le résultat change, et ce modèle ne peut alors être équilibré. Certains économistes peuvent s'opposer à cette hypothèse de non-équilibre. La doctrine standard affirme que le non-équilibre ne peut être important dans l'économie. L'équilibre est l'état naturel de l'économie sur les théories économiques traditionnelles. Toutes les hypothèses mathématiques sont basées sur l'équilibre du marché. W. BRIAN ARTHUR argue que ce n'est pas le cas. Il considère que le déséquilibre est l'état naturel de l'économie, car cette dernière est toujours ouverte à la réaction à cause des chocs extérieurs et du non-équilibre endogène par les interactions des agents. Selon lui, deux raisons expliquent cela : l'une est l'incertitude fondamentale, l'autre est l'innovation technologique (ARTHUR, 2013). Dans le non-équilibre, le processus de l'économie n'est plus certain, prévisible et mécanique, il devient dépendant, organique et évolutionnaire. L'économie peut parfois être simplifiée à un équilibre standard avec homogénéité, mais ce n'est pas souvent le cas, elle est en évolution constante

et montre toujours un nouveau comportement. Dans les études statiques, les problèmes des choix de l'équilibre et de la prévision sont incertains. La présence de ces incertitudes n'est cependant pas surprenante, parce que celle-ci sont essentiellement des problèmes émergents, qui ne peuvent être résolus par l'analyse statique. Ces deux types de problèmes apportent beaucoup de choses insatisfaisantes pour les économistes. Mais, dans les études sur le déséquilibre, comme le modèle *basé agents*, ces deux types de problèmes ont été correctement placés, et même les incidents ont été résolus.

Chapitre 3

Anomalies sur les marchés financiers

3.1 Introduction

Avant les années 1980, la théorie financière classique occupait une position dominante dans la théorie financière, y compris la théorie du portefeuille de MARKOZITZ (modern portfolio theory, *MPT*, 1952), le modèle d'évaluation des actifs financier de SHARPE, LINTNER et MOSSIN (capital asset pricing model, *CAPM*, 1964, 1965 et 1966), le modèle d'évaluation par arbitrage du ROSS (arbitrage pricing theory, *APT*, 1976), la théorie d'évaluation des options de BLACK-SCHOLES et MERTON (option pricing theory, *OPT*, 1973), etc. Toutes ces théories ont hérité des hypothèses de base de l'économie néo-classique comme les marchés parfaits, les gens rationnels, entre autres dont la pierre angulaire est l'hypothèse du marché efficient (*Efficient Markets Hypothèse, EMH*), et l'objet de l'étude est le comportement des marchés financiers dans l'hypothèse de marchés parfaits et d'individus rationnels.

La théorie de la finance moderne est fondée sur l'hypothèse du marché efficient et la théorie des attentes rationnelles. L'hypothèse du marché efficient suppose que le prix actuel reflète toutes les informations disponibles. Les informations sur les prix historiques

n'ont pas d'effet pour prévoir les prix futurs, la volatilité et les risques du marché sont exogènes. Le prix d'équilibre du marché est décidé par sa valeur fondamentale, et les investisseurs ne peuvent obtenir des profits spéculatifs. Cependant, en réalité, les investisseurs du marché possèdent des informations différentes et leurs interprétations et traitements de l'information sont également différents. Ce ne sont pas des gens rationnels déclarés dans l'hypothèse du marché efficient. Par conséquent, le comportement des prix du marché observé a une différence significative par rapport à celui dans l'hypothèse du marché efficient.

Dans l'hypothèse du marché efficient, étant décidé par sa valeur fondamentale, le prix suit un processus aléatoire. Si l'on suppose que les valeurs fondamentales suivent une distribution normale, alors le ratio du rendement de l'actif sera aussi distribué normalement. La théorie du marché efficient, pierre angulaire de la théorie financière moderne, est largement enseignée, sa place étant profondément enracinée. Recherchée par les économistes, elle a aussi été testée à plusieurs reprises. Mais depuis les années 1980, les recherches empiriques, en contradiction avec l'hypothèse du marché efficient, continuent d'émerger. L'existence des anomalies indique que le marché est invalide, et des anomalies existent dans toute forme de l'hypothèse du marché efficient. Sur les marchés boursiers de nombreux pays, les anomalies sont fréquentes.

L'hypothèse du marche aléatoire ne peut expliquer les faits stylisés sur les marchés financiers réels : le clustering de volatilité, la distribution à queue épaisse, la structure non linéaire de corrélation du rendement, la volatilité du marché, les bulles de marché et l'effondrement du marché. Pendant toute l'année 2015, le marché boursier de la Chine était en proie à l'angoisse, ayant subi de plein fouet une violente crise après plusieurs mois de croissance. Les bourses chinoises ont commencé à monter depuis fin 2014, l'indice ayant atteint un pic le 12 juin ; il avait grimpé de plus de 150% au cours des 12 derniers mois. Le jour même où le marché atteignait son apogée, il a commencé à chuter. Les investisseurs, inquiets, se sont peu à peu désengagés, et le gouvernement était incapable

d'agir, provoquant un plongeon de plus de 30% de la bourse en seulement trois semaines. Personne ne savait où cela allait finir. Bien que le gouvernement chinois ait joué sur le marché boursier, et que la banque centrale et le ministère de la Sécurité publique aient proposé des mesures de sauvetage, la Chine a subi un krach boursier. Comment la bulle s'est-elle formée ? La bulle spéculative est une des anomalies les plus importantes sur le marché financier.

Afin d'expliquer ces anomalies de marché, les chercheurs ont proposé un certain nombre de nouvelles théories et de modèles. En 2003, le prix Nobel a été attribué à deux économètres très connus, ROBERT F. ENGLE et CLIVE GRANGER, qui ont résolu des problèmes d'analyse de séries chronologiques, avec une volatilité variant dans le temps et une non stationnarité. Bien que le modèle ARCH proposé par ENGLE en 1982 ait été utilisé pour la modélisation du taux d'inflation britannique, l'idée a clairement été inspirée par des anomalies du marché financier. Dans les années 1960 et 1970, lorsque l'hypothèse du marché efficient (FAMMA) était populaire, le modèle du marche aléatoire a dû faire face à un défi sans précédent par B.B. MANDELBROT. Dans le modèle du marche aléatoire, la série de rendement était un bruit blanc. Mais MANDELBROT a constaté que la distribution du taux de rendement était à queue épaisse et à pic très pointu, et que la série de rendement présentait également une corrélation à long terme. D'après ces résultats, il a proposé un effet *Noé* et un effet *Joseph* du prix des actions. Selon effet *Noé*, le prix des actions produit de temps en temps un grand saut discontinu et soudain, ce grand saut est toujours suivi par un autre, un petit saut est aussi toujours suivi par un autre. Il s'agit là du phénomène de regroupement de volatilité (*Volatility Clustering*) dans la finance, conduisant à une distribution à queue épaisse du rendement du prix des actions, mais pas à une distribution normale selon l'hypothèse du marché efficient. L'effet *Joseph* se présente quand il y a un phénomène cyclique durable et non périodique pour les prix. Comme l'histoire de *Joseph* dans la Bible, pour la production alimentaire, il s'agit d'une récolte pendant sept ans, suivie d'une famine de sept années, montrant que le rendement a une corrélation à long terme. Les travaux de MANDELBROT ont inspiré le

développement de l'analyse de la série chronologique, et ENGLE a établi le modèle ARCH.

En effet, la plupart des modèles sont basés sur l'hypothèse de la rationalité limitée et sur celle des agents hétérogènes qui supposent que le comportement des investisseurs est une interaction et le marché financier est un système adaptatif complexe qui se développe constamment. Ainsi, l'hétérogénéité prospective des investisseurs peut conduire à une instabilité du marché et à un comportement dynamique complexe. En conséquence, le comportement des prix du marché peut dévier significativement de sa valeur fondamentale. Un des premiers modèles *basés agents* du marché financier, avec des agents hétérogènes, est attribué à ZEEMAN (1974). Le modèle peuplé avec les fundamentalistes et les chartistes explique les phénomènes de commutation dans la proportion des deux types de traders entre les marchés du taureau et de l'ours. Les fundamentalistes et les chartistes représentent deux groupes typiques d'agents au sein de la modélisation financière. Le premier est basé sur leurs décisions concernant les fondamentaux du marché, tels que le rendement du dividende et la croissance économique. Le second est basé sur le modèle historique du prix des actions. Il n'y a aucune règle commune permettant de savoir comment modéliser les fondamentaux et le comportement de la tendance. ZEEMAN a fait valoir que dans le marché du taureau, la proportion des chartistes, qui suivent une tendance à l'année à la hausse allait encore faire monter les prix. La tendance à la hausse se poursuit jusqu'à ce que les fundamentalistes perçoivent que les prix sont trop élevés ; ils commencent alors à vendre, conduisant à une baisse des prix (marché de l'ours) et réduisant la proportion des chartistes respectivement. La baisse provoque les fundamentalistes qui commencent à acheter des stocks, ce qui transforme à nouveau la tendance. Dans les modèles *basés agents*, il existe différents types d'investisseurs en même temps, ayant des philosophies d'investissement différentes et des résultats attendus divers sur la volatilité du futur prix des actifs risqués. Parmi eux, deux types d'investissement sont les plus typiques : une catégorie d'investisseurs est appelée agent de bruit (également nommée analyste graphique ou analyste technique) concernant l'autre catégorie les agents rationnels (ou analystes fondamentaux) pensent que le prix des actifs est déterminé

uniquement par leur valeur fondamentale (*Fundamental Value*) et des outils d'analyse technique pour prédire le prix des actions sont futiles. Dans les modèles *basés agents*, une question est importante : quel est le rôle ou la fonction d'un bruit dans le processus d'évaluation des actifs? En outre, dans l'évolution du prix du marché, comment le bruit (le choc externe) interagit avec le facteur décisif des prix du marché? Il existe trois réponses cette question : (1) Il y a beaucoup de bruits et d'incertitudes dans le marché réel, tous les jours, de nouveaux événements ou informations créant un impact sur le marché financier ; Le bruit ne doit donc pas être ignoré ; (2) Des résultats de certains modèles montrent que certaines interactions et hétérogénéités des investisseurs peuvent conduire à un comportement chaotique pour le marché ; en effet, le comportement dynamique chaotique est extrêmement sensible au bruit ; ainsi, il est nécessaire d'analyser le facteur de bruit et ses effets ; (3) En raison de l'insuffisante attention de la littérature actuelle, il existe peu d'analyses et de recherches détaillées.

Par conséquent, on intègre l'hétérogénéité des investisseurs, l'impact du bruit et les fluctuations du marché dans un même cadre analytique pour examiner les influences des bruits sur l'évolution du marché. Afin d'atteindre cet objectif de recherche, l'analyse traditionnelle ne suffit pas, il faut introduire des idées de modélisation de la finance calculée (*Computational Finance*). À l'aide de la technologie de simulation par ordinateur, on peut décrire les effets du comportement des marchés différents dans le processus de l'évolution du marché global.

Dans ce chapitre, nous tentons d'étudier les phénomènes que les théories classiques de la finance ne permettent pas d'expliquer, les *Faits Stylisés*. CONT (2001) a résumé un ensemble de faits stylisés empiriques issus de l'analyse statistique des variations de prix dans divers types de marchés financiers. Nous présentons ensuite trois exemples des faits stylisés du marché financier : la queue épaisse, la volatilité clustering et la mémoire longue. Ces éléments peuvent-ils être reproduits facilement grâce à des simulations *basées agents*? Nous allons proposer un modèle d'implémentation sous la forme

d'une plate-forme de simulation de marché réalisée en langage C++.

3.2 Queue épaisse

Dans la théorie classique, les changements de prix des titres suivent une marche au hasard. C'est-à-dire, les rendements des cours de bourses sont distribués selon une loi normale de moyenne nulle et de variance σ^2 . Autrement dit, les évolutions des prix sont indépendantes des prix passés. On ne peut prévoir les prix futurs parce qu'ils sont totalement aléatoires. En économétrie, une variété de moyens permet de voir si c'est une distribution normale. Le test *Jarque-Bera* qui utilise uniquement l'asymétrie et le *Kurtosis*, est le plus utilisé. On prend un estimateur non biaisé pour la loi normale du *Kurtosis* normalisé comme plusieurs logiciels de statistiques (*SAS, Excel, LibreOffice Calc*, etc). La valeur de *Kurtosis* est calculée comme suit :

$$K = \frac{T(T+1)}{(T-1)(T-2)(T-3)} * \frac{\mu^4}{\sigma^4} - 3 * \frac{(T-1)^2}{(T-2)(T-3)} \quad (3.1)$$

où μ^4 est le quatrième moment central et σ est l'écart type. On prend T comme nombre de périodes, parce que N correspond au nombre d'agents dans notre modèle. Si le rendement suit une distribution normale, la valeur de *Kurtosis* doit être nulle. Pourtant, un grand nombre de faits montre que, sur le marché financier réel, la valeur de *Kurtosis* est toujours supérieure à 0. La courbe de répartition du rendement est une caractéristique de la queue épaisse. C'est-à-dire, les probabilités du rendement élevé et du rendement faible sont supérieures à celles de la loi normale. JONDEAU et ROCKINGER (2003) ont trouvé que, dans un marché financier mature, la queue de gauche était plus lourde que celle de droite. GRAY et FRENCH (1990) ont discuté la distribution de puissance exponentielle, laquelle a un pic et une queue épaisse ; la queue étant réduite à un taux exponentiel, il est possible de donner un bon ajustement du rendement.

3.2.1 Les données

L'ensemble des données que nous allons analyser dans cet article est l'indice des prix de clôture quotidien du marché boursier français *CAC 40*. Avec 30 ans d'historique et environ 247 jours d'ouverture par an, il existe 7410 fluctuations, du 31 décembre 1987 (jour de lancement sur une base de 1000 points) au 31 mars 2017. On définit P_t comme l'indice du prix du *CAC 40* au moment t ($t=0,1,2,\dots,7410$). Alors le rendement R_t est :

$$R_t = \text{LN}\left(\frac{P_t}{P_{t-1}}\right) \quad (3.2)$$

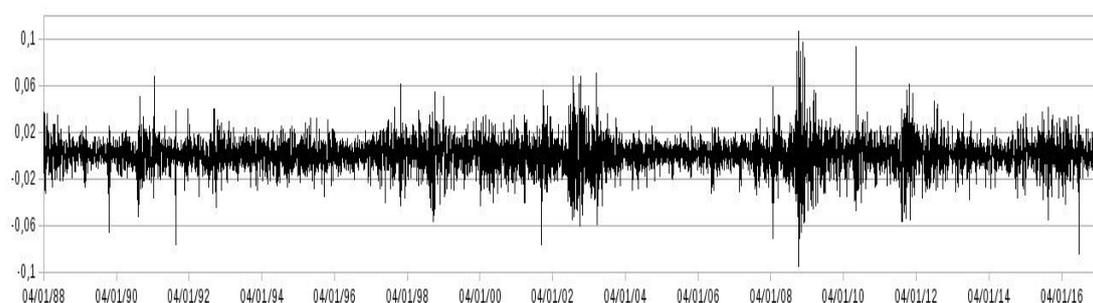
où $t = 1,2,\dots,7411$. Nous prenons le rendement logarithmique dans cette analyse empirique du rendement des prix du marché financier quotidien, car la caractéristique du rendement logarithmique apporte un certain confort pour le calcul et la modélisation. Il y a quatre avantages : 1) le niveau des log permet de lisser la série, mais il ne change pas la relation entre les données. Il affaiblit aussi l'hétéroscédasticité et la colinéarité des données. Il est facile à calculer ; 2) Il est favorable d'observer directement le rendement des actions, car le taux de rendement est un concept d'intérêt composé. Si l'on prend les variables elles mêmes, il est difficile de voir les variations des rendements ; 3) il comporte un avantage d'analyse non seulement latérale mais aussi verticale. En tant qu'intérêt composé, c'est aisé de comparer les indices des deux marchés ou les indices des deux périodes ; 4) toutes les données transformées en log sont positives, ce qui est favorable à la régression linéaire.

Les figures 3.1 et 3.2 présentent les cours historiques et leurs rendements journaliers du *CAC 40*.

FIGURE 3.1 : Indice des prix journaliers du CAC 40 31/12/1987-31/03/2017



FIGURE 3.2 : Rendements des prix journaliers du CAC 40 31/12/1987-31/03/2017



3.2.2 La distribution des rendements du CAC 40

Au cœur de la théorie financière, l'hypothèse est que les rendements journaliers des actifs suivent une loi normale de moyenne μ et d'écart type σ . Mais beaucoup de faits montrent que les distributions de rendements des actifs financiers ont des queues plus épaisses que les distributions normales. MANDELBROT a remarqué pour la première fois ce phénomène en analysant les cours du coton en 1963. Nous allons d'abord vérifier si la courbe de rendement du CAC 40 représente la distribution gaussienne.

La table 3.1 donne des statistiques sommaires pour le rendement R_t .

TABLE 3.1 : Statistiques sommaires pour le rendement R_t

data	nombre	min	max	moyen	écart type	asymétrie	kurtosis	gamme étudiée
R_t	7410	-0.095	0.106	0.0002	0.014	-0.086	4.583	14.6

Dans la table 3.1, on voit clairement que la valeur de *Kurtosis* est de 4,58, étant supérieure à celle de la distribution normale qui équivaut à 0. Cette valeur n'est pas très

élevée par rapport à celle sur le *Standard & Poor 500*, qui est de 25,7, mais elle montre quand même un pic. La valeur de la gamme étudiée est de 14,6, qui est la gamme divisée par l'écart type. Cette valeur indique que les données de R_t illustrent un degré de dispersion relativement important. La valeur de *Kurtosis* et la valeur de la gamme étudiée montrent une caractéristique de la queue épaisse par rapport à une distribution normale. La valeur d'asymétrie est négative, soulignant que la queue de gauche est plus longue et que la valeur de la moyenne est à gauche de la valeur du pic. C'est-à-dire, la continuité du rendement négatif est supérieure à celle du rendement positif. Nous revoyons la valeur de la moyenne 0,000 22 et celle de la médiane 0,000 42, étant supérieures à 0. Autrement dit, l'échantillon est rentable. La valeur de la moyenne est inférieure à celle de la médiane ; cela indique aussi que la distribution de rendement a une queue gauche longue. La valeur du maximum est environ de 0,1, celle du minimum est environ de -0,1, et la valeur de l'écart type est aussi proche de 0,1, montrant que le bénéfice net par action comporte de modestes différences. Toutes ces valeurs sont conformes aux caractéristiques des gains robustes. Nous pouvons affirmer que le *CAC 40* est un marché mature. Son rendement a une queue épaisse, mais pas très forte.

$$A = \frac{T}{(T-1)(T-2)} * \frac{\mu^3}{\sigma^3} \quad (3.3)$$

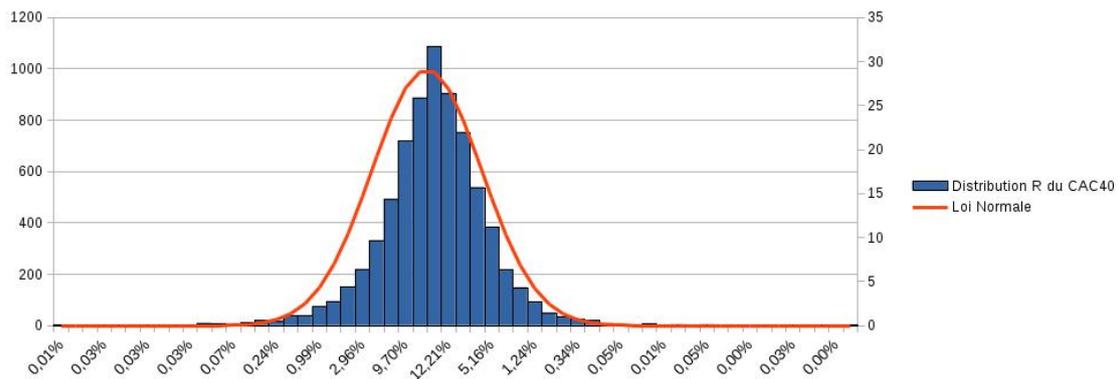
$$G = \frac{max - min}{\sigma} \quad (3.4)$$

Ces deux équations montrent comment calculer la valeur d'asymétrie et la valeur de la gamme étudiée. Le μ^4 est le quatrième moment central. Le programme 1¹ montre la façon dont nous avons réalisé les statistiques sommaires pour le rendement R_t en C++.

Quand nous regardons la figure 3.3, nous voyons que l'allure de courbe en cloche est

1. voir Annexes

FIGURE 3.3 : Distribution du rendement du CAC 40 et distribution normale



trompeuse, parce que les observations du *CAC 40* ne sont pas du tout gaussiennes. Bien évidemment, beaucoup de données ne se trouvent pas dans la zone de la loi normale. Le *CAC 40* contient de nombreux événements extrêmes (en positif et en négatif). La distribution de ses fluctuations appartient au régime des distributions, appelées à queue épaisse. La queue de la distribution n'est pas négligeable, et des événements extrêmes n'y sont pas improbables.

3.3 Mémoire à long terme et volatilité clustering

Comme nous le savons, si un marché de capitaux est efficace, alors le rendement du marché R_t lui-même a presque peu de corrélation sérielle ; les variations des cours peuvent être considérées comme un processus de marche aléatoire. TAYLOR (2008) a constaté que la série du rendement absolu $|R_t|$ avait une auto-corrélation évidente, surtout pour un grand lag, la série ayant un coefficient d'auto-corrélation positif, malgré tout. DING, GRANGER et ENGLE (1993) ont analysé les rendements du *Standard & Poor 500* et observé que la série $|R_t|^d$ ($d > 0$) avait une forte auto-corrélation. Quand $d = 1$, le coefficient d'auto-corrélation de la série est le plus élevé. À ce moment-là, le lag de la série est supérieur à 2500. Cela montre une forte mémoire à long terme. Nous allons réaliser une auto-corrélation et une analyse de corrélation croisée pour le marché financier français

3.3.1 Définition de la mémoire longue

Selon la définition de la mémoire à long terme pour une série donnée par GRANGER et DING (1996), on suppose une série de constantes $\{h_j\}, j = 0, 1, 2, \dots$, puis on définit une série de filtres en changement, $h_t(B) = \sum_{j=0}^t h_j B^j$, où B est un opérateur inverse. On suppose une série $\{e_t\} : e_t = m + \epsilon_t$, où ϵ_t est un bruit blanc avec une moyenne zéro et une variance constante σ_0^2 , la covariance $\text{corr}(\epsilon_t, \epsilon_s) = 0, t \neq s$. Par la suite, la série $\{e_t\}$ génère une série $\{x_t\}, x_t = h_t(B)e_t$, alors que $m(t) = mh_t(1), v(t) = \sigma_0^2 \sum_{j=0}^t h_j^2, s(t) = \sum_{j=0}^t h_j \epsilon_{t-j}$, et

$$f_N(\omega) = E[N^{-1} | \sum_{t=0}^N S_t e^{it\omega} |^2] \quad (3.5)$$

Pour tous les $\omega \geq 0$, si $f(\omega) = \lim_{N \rightarrow \infty} s_t e^{it\omega}$ existe, alors $f(\omega)$ est appelé spectre de x_t . Si $\{h_t(1)\}$ est une série monotone et croissante, alors $m(t)$ et $v(t)$ sont appelés tendance de la moyenne et variance de x_t . Si le spectre de la série $\{x_t\}$ possède les propriétés suivantes, cette série $\{x_t\}$ est appelée série à longue mémoire :

1) quand $\omega \rightarrow 0, f(\omega) \rightarrow \infty$

2) presque pour toutes les valeurs de $\omega, f(\omega)$ a une borne supérieure, c'est à dire que $f(\omega)$ n'a pas de borne supérieure pour seulement quelques valeurs de ω .

Pour la propriété 1), une condition nécessaire est établie : la série $\{h_j\}$ est une série divergente. Si $h_j = A_j^{d-1}$, alors que la série générée par la série $\{h_j\}$ est un processus intégrable. Pour les valeurs de ω les plus faibles, le spectre de ce processus est proportionnel à la valeur ω^{-2d} . Si le d est un non entier, ce processus est un procédé intégrable fractionnaire. Par rapport au procédé ARMA stable et standard (pendant ce processus,

l'atténuation de h_j suit la puissance θ^j , $|\theta| < 1$), un processus est connu comme ayant une mémoire longue, parce qu'une vibration ϵ_t au moment t peut continuer à affecter des valeurs de x_{t+k} dans une longue période au future.

Le concept de la mémoire longue a été initialement développé par HURST (1951). La mémoire à long terme est définie comme une série ayant un corrélogramme qui diminue lentement ou de manière équivalente, un spectre infini à la fréquence zéro. C'est-à-dire, que la mémoire longue peut être définie en termes de taux de désintégration des auto-corrélations à long lag. En particulier, un processus stationnaire a une longue mémoire (ou une dépendance à la gamme longue). S'il existe un nombre réel d et une constante $c_\rho > 0$, alors :

$$\lim_{k \rightarrow \infty} \frac{\rho(k)}{c_\rho k^{2d-1}} = 1 \quad (3.6)$$

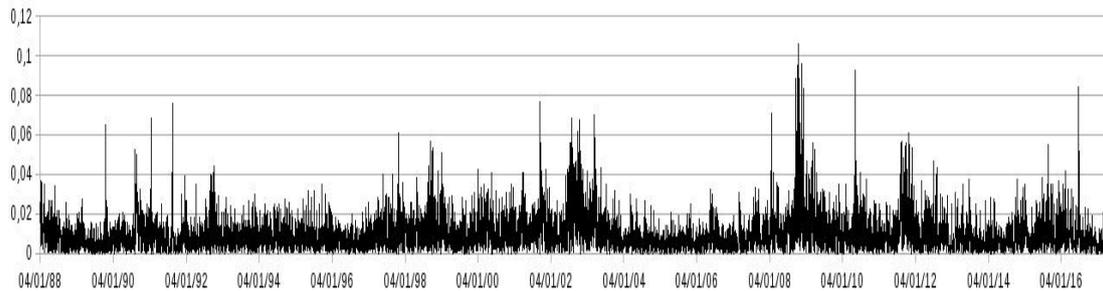
où ρ_k est l'auto-corrélation au lag k et d est le paramètre de la mémoire longue. Les auto-corrélations d'un processus à mémoire longue ne sont pas sommables. Mais DING, GRANGER et ENGLE (1993) ont réussi à comparer et à examiner les auto-corrélations de r_t , $|r_t|^d$ ($d > 0$) sur le *Standard & Poor 500*. Poursuivant ce point, nous allons examiner les auto-corrélations de r_t , $|r_t|^d$ ($d > 0$) sur le marché financier français du *CAC 40*. Nous allons tenter de les reproduire en C++.

3.3.2 Analyse de la mémoire longue du CAC 40

En 1986, Taylor a trouvé que la série $|r_t|$ avait une corrélation sérieuse positive significative sur de longs délais, alors que $|r - t|^d$ peut être considéré comme ayant une mémoire longue. Quand $d = 1$ ou est proche de 1, cette propriété est plus forte par rapport aux autres valeurs de d . Dans cette section, nous allons faire des analyses sur des auto-corrélations et des corrélations croisées sur le *CAC 40*. La figure 3.4 donne les historiques

du $|r_t|$ du CAC 40 pendant 30 ans.

FIGURE 3.4 : Rendements absolus des prix journaliers du CAC 40 31/12/1987-31/03/2017.



D'après les figures 3.2, 3.3 et 3.4, nous voyons clairement le mouvement à long terme des p_t , r_t et $|r_t|$ au cours des 30 dernières années. Pour le prix quotidien p_t , il y a une tendance à la hausse dans la première dizaine d'années, puis une fluctuation répétée dans les vingt années suivantes. Mais la série du rendement r_t est plus stable autour de la moyenne $\mu = 0,00022$.

Analyse des auto-corrélations pour r_t , $|r_t|$ et r_t^2 Dans beaucoup de modèles financiers théoriques, le rendement d'un actif financier suit un *i.i.d* procédé (indépendant et identiquement distribué), les rendements étant des variables aléatoires *i.i.d*. Mais, le prix d'un actif peut être modifié à mesure de l'arrivée des informations, si le marché est efficient. S'il y a une grappe d'informations qui arrivent en même temps, la distribution du rendement au moment $t+1$ peut dépendre de celle du rendement du moment t , bien que les rendements soient des variables aléatoires *i.i.d*. Alors, est-ce que ces variables n'ont pas du tout de corrélation comme on le suppose dans l'hypothèse du marché efficient? DING, GRANGER et ENGLE (1993) ont obtenu la conclusion contraire lorsqu'ils ont analysé les prix du *Standard & Poor 500*. Nous allons examiner le CAC 40 pour vérifier si nous pouvons avoir le même résultat.

La table 3.2 montre des auto-corrélations pour r_t , $|r_t|$ et r_t^2 au lag de 1 à 5 et 10, 20, 70, 100. Selon les valeurs des auto-corrélations, nous savons que l'auto-corrélation

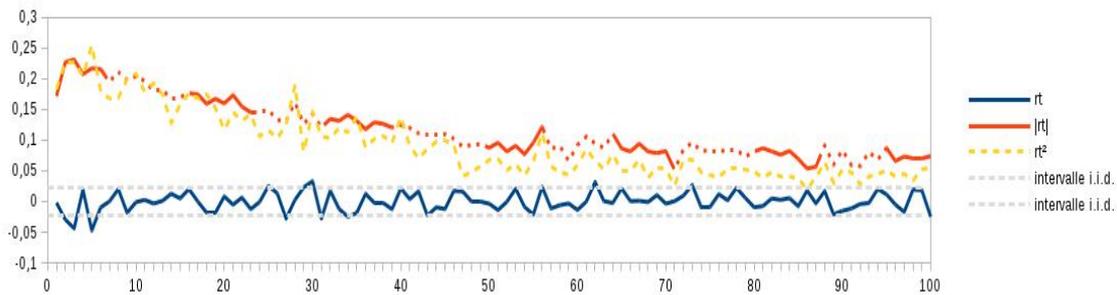
TABLE 3.2 : Auto-corrélations pour le rendement r_t

data	lag 1	lag 2	lag 3	lag 4	lag 5	lag 10	lag 20	lag 70	lag 100
r_t	-0.002	-0.030	-0.044	0.019	-0.047	-0.001	0.008	-0.004	-0.025
$ r_t $	0,172	0,227	0,231	0,208	0,217	0,205	0,160	0,082	0,074
r_t^2	0,183	0,226	0,228	0,205	0,255	0,208	0,117	0,053	0,055

du rendement r_t du CAC 40 est très faible. Parmi les 5 premiers lags, quatre sont négatifs. En particulier, le coefficient d'auto-corrélation au premier lag est $-0,002$ qui ne correspond pas aux recherches de MALKIEL et FAMA (1970), TAYLOR (2008), où les chercheurs ont trouvé que la majorité des séries du rendement sur le marché financier avaient un très faible coefficient d'auto-corrélation positif au premier lag. Évidemment, le marché financier français ne fait pas partie de la majorité. Il est particulier. Ces coefficients d'auto-corrélation des 3 premiers lag sont tous négatifs, les changements de prix suivent un procédé de marche aléatoire, pouvant signifier que le rendement du CAC 40 n'a pas du tout de mémoire. Dans l'hypothèse de l'efficience du marché financier, si la série du rendement r_t est un procédé *i.i.d*, alors toute transformation de cette série r_t est également un processus *i.i.d*, leur coefficient devant être 0. Mais, dans la table 3.2, les auto-corrélations des séries $|r_t|$ et r_t^2 sont très fortes. Même quand le lag est supérieur à 100, le coefficient d'auto-corrélation est toujours positif. Les coefficients d'auto-corrélation de la série $|r_t|$ sont plus élevés que ceux de la série r_t^2 . C'est-à-dire, la mémoire longue de la série $|r_t|$ est plus forte que celle de la série r_t^2 . Cela montre que r_t n'est pas un procédé *i.i.d*. Le rendement r_t a une mémoire, dans une certaine mesure. Bien que cette mémoire soit faible, il y a encore une petite fraction de rendements des actions qui sont prévisibles.

La figure 3.5 montre aussi ce que nous avons analysé précédemment. Nous voyons clairement que la majorité des auto-corrélations de la série r_t se trouvent dans un intervalle, entre les deux lignes pointillées en gris. Les valeurs de ces deux lignes sont $\pm 1.96/\sqrt{T}$. Cela signifie que si r_t est un procédé indépendant et identiquement distribué, 95% des valeurs d'auto-corrélation de la série r_t doivent être dans cet intervalle de

FIGURE 3.5 : Auto-corrélations pour r_t , $|r_t|$ et r_t^2



confiance. BARTLETT (1946) a prouvé que si r_t est un procédé i.i.d., alors son coefficient d'auto-corrélation ρ_t est d'environ $N(0,1/T)$, où T est la taille de l'échantillon. Pour le rendement du CAC 40, $T = 7410$, alors le seuil = $\pm 1.96/\sqrt{7410}$, soit $\pm 0,023$.

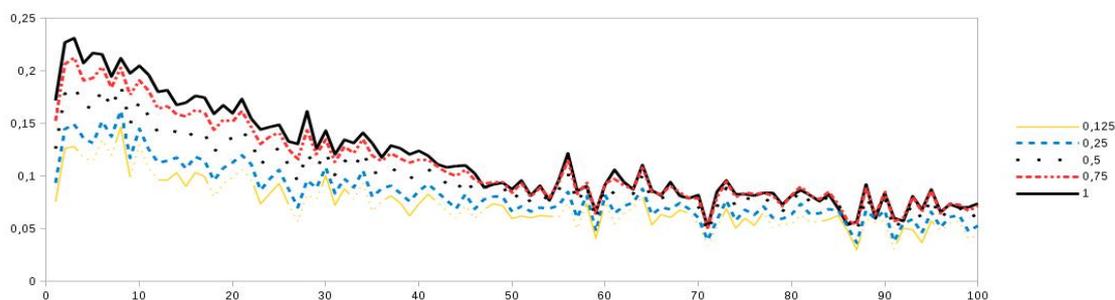
Analyse des auto-corrélations pour $|r_t|^d$ Concernant la forme de transformation de la série du rendement absolu du CAC 40 $|r_t|^d$, pour différentes valeurs positives de d , celles-ci ont toutes une auto-corrélation positive significative. Les lags de cette auto-corrélation sont supérieurs à 100. Dans la table 3.3, nous avons le résultat des auto-corrélations de la série $|r_t|^d$ avec différents $d = 0,125, 0,25, 0,5, 0,75, 1, 1,25, 1,5, 1,75, 2, 3$. au lag de 1 à 5, 10, 20, 70 et 100. La figure 3.6 montre l'auto-corrélogramme du $|r_t|^d$ au lag de 1 à 100, pour $d = 0,125, 0,25, 0,5, 0,75$ et 1. La figure 3.7 souligne l'auto-corrélogramme de $|r_t|^d$ au lag de 1 à 100, pour $d = 1,25, 1,5, 1,75, 2$ et 3. Toutes les transformations de puissance du rendement absolu ont des auto-corrélations positives significatives jusqu'au lag 100. Cela indique que les rendements du marché boursier ont une mémoire à long terme. À partir de la table 3.2, nous pouvons argumenter que le coefficient d'auto-corrélation est augmenté progressivement au début, à mesure de l'augmentation de la valeur d ; lorsque la valeur de d est autour de 1,5, le coefficient d'auto-corrélation est plus grand, puis il diminue progressivement. Cette conclusion ne correspond pas à celle de DING, GRANGER et ENGLE (1993). Dans leur recherche, lorsque le coefficient d'auto-corrélation du $|r_t|^d$ est le plus grand, c'est quand $d = 1$.

Correspondant à la table 3.2, la figure 3.8 donne la relation des coefficients d'auto-

TABLE 3.3 : Auto-corrélation pour $|r_t|^d$

d	lag 1	lag 2	lag 3	lag 4	lag 5	lag 10	lag 20	lag 70	lag 100
0,125	0,076	0,126	0,128	0,118	0,114	0,129	0,099	0,054	0,044
0,25	0,093	0,145	0,149	0,136	0,131	0,145	0,113	0,060	0,053
0,5	0,126	0,178	0,185	0,167	0,164	0,171	0,136	0,071	0,065
0,75	0,152	0,206	0,213	0,191	0,193	0,191	0,152	0,079	0,072
1	0,172	0,227	0,231	0,208	0,217	0,205	0,160	0,082	0,074
1,25	0,184	0,239	0,241	0,217	0,235	0,213	0,159	0,080	0,072
1,5	0,189	0,242	0,243	0,219	0,247	0,216	0,150	0,073	0,068
1,75	0,188	0,237	0,238	0,214	0,253	0,214	0,135	0,064	0,062
2	0,183	0,226	0,228	0,205	0,255	0,208	0,117	0,053	0,055
3	0,149	0,163	0,166	0,147	0,245	0,170	0,049	0,017	0,029

FIGURE 3.6 : Auto-corrélation de $|r_t|^d$, $d = 0.125, 0.25, 0.5, 0.75, 1$



corrélation et d pour des lags 1, 2, 7 et 10. La figure 3.8 montre la même description que la table 3.2. Avec l'augmentation de la valeur de d , le coefficient d'auto-corrélation augmente progressivement, après avoir atteint une valeur maximale, et même si la valeur de d continue à augmenter, le coefficient d'auto-corrélation diminue progressivement. Pour des coefficients des lags 1, 2, 7 et 10, quand $d = 1,5, 1,5, 1,25$ et $1,5$, le coefficient d'auto-corrélation est le plus grand. Pour différentes valeurs de d , c'est le coefficient d'auto-corrélation au lag 2 qui est le plus grand.

Grâce à ces données, nous trouvons que la série de $|r_t|^d$ maintient une auto-corrélation positive supérieure à 100 lag. Ainsi, quand le coefficient d'auto-corrélation de $|r_t|^d$ donne pour la première fois une valeur négative ? La table 3.4 montre le résultat.

TABLE 3.4 : Auto-corrélation pour $|r_t|^d$

d	0.125	0.25	0.5	0.75	1	1.25	1.5	1.75	2	3	5	10
τ^*	231	231	278	306	242	242	240	233	233	124	29	14

FIGURE 3.7 : Auto-corrélation de $|r_t|^d$, $d = 1.25, 1.5, 1.75, 2, 3$

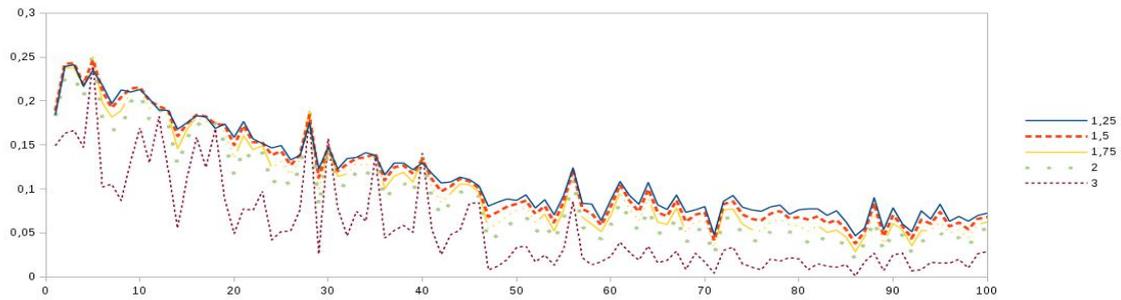
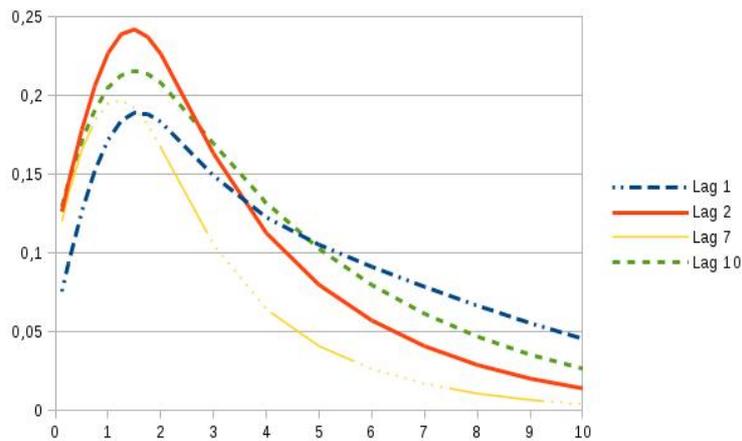


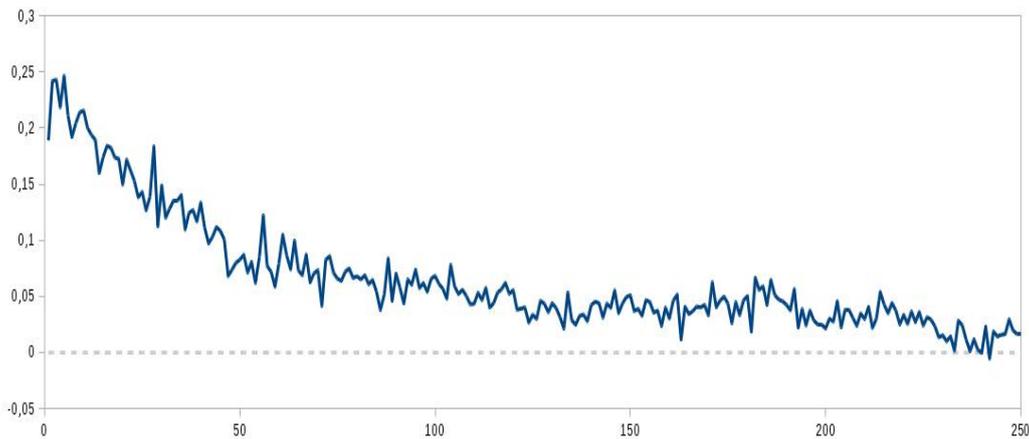
FIGURE 3.8 : Auto-corrélation de $|r_t|^d$ au lag 1, 2, 7 et 10



Sur la figure 3.5, nous voyons que les coefficients d'auto-corrélation de la série r_t sont très petits pour chaque lag, et la majorité est située dans l'intervalle de confiance 95%. Mais pour la série $|r_t|^d$, la situation est tout à fait différente. Non seulement les coefficients d'auto-corrélation de la série $|r_t|^d$ sont tous positifs avant le lag 100, mais ils diminuent aussi très rapidement pour les premiers 50 lags, et après le lag 50, ils se dégradent lentement. La série $|r_t|^d$ a une auto-corrélation positive supérieure à 250 lag. Ce chiffre est presque égal au nombre de jours d'ouverture annuelle du CAC 40, soit 247 jours par an comme calculé plus haut.

L'analyse à la longue mémoire sur les rendements du marché boursier français prouve que le rendement du CAC 40 a une mémoire à long terme, mais elle n'est pas aussi forte que celle du marché américain, *Standard & Poor 500*.

FIGURE 3.9 : Auto-corrélation de $|r_t|^d$, $d = 1.5$, jusqu'au lag 250



3.4 Autres anomalies sur le marché financier

3.4.1 Les effets sur la petite entreprise

Sur le marché boursier, un des premiers cas qui révèle que le marché efficient n'est pas efficace est l'effet de taille des entreprises. En statistique, un effet de taille est la mesure de la force de l'effet observé d'une variable sur une autre, et plus généralement d'une inférence.

Ce sont des anomalies causées par des niveaux de reconnaissance de la société elle-même ou des investisseurs visant la société. BANZ (1981) a rapporté que les petites entreprises avaient des rendements ajustés au risque considérablement plus élevé que les grandes, et il a nommé cette constatation "*Effet Taille*". C'est la première étude empirique connue qui présente des preuves d'un effet taille. Le chercheur a analysé toutes les actions ordinaires cotées à la *Bourse de New York* sur la période 1936-1975. Selon la taille des sociétés cotées, il a divisé la totalité du stock de la *Bourse de New York* en cinq groupes. Il a trouvé que le rendement moyen du stock avec la taille la plus petite était 19,8% plus élevé que celui du stock avec la taille la plus grande. Il existe un phénomène de corrélation négative entre ce rendement boursier et la taille des sociétés cotées. C'est-à-dire, le

coefficient sur la valeur de marché est négatif et significatif, ce qui indique que les petites entreprises ont des rendements ajustés au risque plus élevé que les grandes entreprises. Ce résultat est robuste par rapport aux changements dans la définition du portefeuille de marché. L'effet de taille varie considérablement au fil du temps ; cependant, il assume une valeur négative sur la période de 1946 à 1955. En outre, l'effet de taille n'est pas linéaire, étant plus prononcé pour les plus petites entreprises dans l'échantillon. BANZ affirme : « *...the size effect exists, but it is not at all clear why it exists. Until we find an answer, it should be interpreted with caution.* » Il a conjecturé qu'une incertitude plus élevée, en raison d'informations insuffisantes sur les petites entreprises, pouvait provoquer un effet de taille.

Beaucoup d'études montrent que, à long terme, même en tenant compte de leur énorme risque, les investissements dans les petites entreprises reçoivent également des profits excessifs. Au cours des dernières années, cet effet s'est affaibli, mais il y a toujours un grand défi pour l'hypothèse du marché efficient. En même temps, de nombreux chercheurs pensent que l'effet sur la petite entreprise est attribué aux éléments suivants : la restructuration des rapports des investisseurs institutionnels, les considérations fiscales, la liquidité plus faible des actions des petites entreprises, la grande dépense d'évaluation des petites entreprises, des règles inappropriées de mesure de risque, etc.

REINGANUM (1981) a également constaté que le rendement moyen des entreprises de taille plus petite était plus élevé que le rendement théorique prévu selon la théorie CAPM. AMIHUD et MENDELSON (1986) ont proposé une théorie de "prime de liquidité" pour expliquer l'effet taille. Selon eux, le volume de négociation des petites entreprises est plus faible par rapport à celui des grandes, qui impacte la liquidité de l'action de l'entreprise. Il faut donc qu'une prime de revenus existe afin de compenser cela pour le rendement de l'action des petites entreprises. LAKONISHOK, SHLEIFER et VISHNY (1994) ont révélé que les actions avec un bénéfice net élevé avaient un risque supérieur, surtout dans la baisse du marché et la récession économique, où leurs performances sont particulière-

ment pauvres. La relation inverse entre le ratio du bénéfice et le taux de rendement est un défi pour l'hypothèse du marché efficient, car à ce moment-là, l'information connue a des prévisions significatives pour le taux de rendement. SIEGEL (1998) a constaté que le taux de rendement annuel des actions à petite capitalisation était supérieur de 4,7% par rapport à celui des actions à grande capitalisation, et que l'effet taille des petites entreprises était surtout concentré en janvier.

3.4.2 Les effets de calendrier

Le rendement d'une action est lié au temps, ce qui signifie qu'il y a une différence systématique du rendement de l'investissement à divers moments. Une collection de théories variées affirment que certains jours, mois ou périodes de l'année sont soumises aux fluctuations de prix supérieures à la moyenne des indices du marché, ne constituant donc pas des moments bons ou mauvais pour investir. On appelle cela les effets de calendrier. Les effets de calendrier comprennent *l'effet Janvier*, *l'effet Lundi*, *l'effet d'Halloween* ou *l'effet Octobre*. Parmi eux, *l'effet janvier* est le plus important.

La découverte de *l'effet Janvier* est attribuable à un économiste américain WACHTEL (1942). Il a publié une autre étude montrant que les actions du *Dow Jones Industrials* de 1927 à 1942 présentaient une saisonnalité haussière marquée, surtout de décembre à janvier. En 1976, deux économistes, ROZEFF et KINNEY (1976) ont constaté qu'entre 1904 et 1974, le rendement de l'indice boursier de *New York Stock Exchange* en Janvier était significativement plus élevé que celui des autres 11 mois. GULTEKIN et GULTEKIN (1983) a étudié les taux de rendement de 17 pays entre 1959 et 1979, et a aussi trouvé que le taux de rendement en janvier était plus élevé que les autres 11 mois dans 13 pays. Selon les données statistiques des actions de *New York Stock Exchange*, pendant les 70 dernières années, le taux de rendement moyen en janvier est de 3,48% et le taux de rendement moyen des autres mois est de 0,42%. RENLIANG ZHANG et BIN HU (1997)

ont étudié le marché boursier de Hong Kong, et ont souligné que l'effet janvier était le plus évident. NAN LU (2012) a choisi au hasard 18 actions du marché boursier de Shanghai comme échantillon typique pour faire une recherche empirique. Il a procédé à une analyse et à des statistiques sur le rendement journalier et son résidu des deux actions montre que la Bourse de Shanghai a un effet de calendrier évident.

3.5 La mesure de reproduction des anomalies

Dans les sections ci-dessus, nous avons mis en évidence les faits stylisés des rendements boursiers. CONT (2001) a présenté un ensemble de faits statistiques issus de l'étude empirique du rendement des actifs, mais il a dit que la plupart des modèles actuellement existants ne pouvaient reproduire en même temps toutes ces caractéristiques statistiques. En effet, il a ignoré le modèle *basé agents*. Le modèle *basé agents* est capable de reproduire certaines des propriétés statistiques des rendements observés sur les marchés boursiers réels. LEBARON (2006) a résumé quelques-uns des résultats du modèle *basé agents* et trouvé qu'ils pouvaient être adaptés à une large gamme de caractéristiques empiriques des marchés financiers. La plupart des modèles *basé agents* sur le marché financier peuvent combiner facilement un bon sous-ensemble de faits stylisés. CHEN, CHANG et DU (2012) ont aussi montré que les faits stylisés pouvaient être reproduits qualitativement par différentes classes du modèle *basé agents*. Ces faits comprennent la répartition des rendements qui est plus élevée que la distribution gaussienne, que les périodes de volatilité persistante, les périodes de volume d'activité persistante et la corrélation entre la volatilité et le volume de transaction.

En général, le modèle *basé agents* peut être validé si les propriétés statistiques des données simulées correspondent à celles des données réelles. Nous pouvons dire que les deux séries peuvent appartenir à la même distribution. Le but ultime de cette thèse est d'expliquer que notre modèle *basé agents* peut simuler des faits stylisés sur le marché

financier français *CAC 40*. Mais avant d'entrer réellement dans notre modèle, beaucoup d'autres éléments doivent être préparés. Par exemple, comment faire le meilleur choix de l'ensemble des paramètres reproduisant des données réelles ?

ALFARANO, LUX et WAGNER (2005) ont estimé directement les paramètres sous-jacents en utilisant une approche paramétrique. Ils ont montré que les paramètres du modèle pouvaient être estimés en appliquant la méthode du maximum de vraisemblances, en 2005, afin de mieux connaître les origines comportementales potentielles des régularités statistiques des données.

GILLI et WINKER (2003) ont introduit une approche d'estimation indirecte, basée sur la comparaison des moments des données réelles et des moments simulés du modèle *basé agents*. Cela implique la minimisation d'une approximation stochastique d'une fonction objective $f(\theta)$. Afin de capturer les principales caractéristiques des données, la fonction objective doit être un mélange de moments supérieurs et d'autres statistiques.

FABRETTI (2013) a donné un étalonnage fournissant des critères pour le meilleur choix des paramètres définis en reproduisant des données réelles. Il a utilisé l'algorithme *Nelder-Mead* simple couplé à *Threshold Accepting* et introduit le modèle *Kirman* qui a été étalonné à l'aide d'une fonction objective combinant des erreurs d'estimation sur le *Kurtosis* et l'effet *ARCH(1)*. Mais son résultat n'est pas très convaincant. Les données simulées montrent moins de queue épaisse et moins de volatilité clustering. La fonction objective est naïve, le *Kurtosis* et l'effet *ARCH(1)* simulés ne permettant pas de reproduire des propriétés statistiques et leur poids a été arbitrairement défini.

WINKER, GILLI et JELESKOVIC (2007) ont indiqué que la fonction objective était une combinaison d'erreurs d'estimation sur certaines quantités choisies (moments ou statistiques). Le choix des moments et des statistiques est très important. Ces derniers doivent être robustes, refléter les propriétés statistiques des données et présenter le potentiel de discrimination entre les modèles alternatifs ou les valeurs des paramètres. L'estimation

peut être effectuée à l'aide du *Moving Block Bootstrap*.

3.5.1 Le *Moving Block Bootstrap*

L'origine du *bootstrap* Des bottes hautes peuvent avoir une languette, une boucle ou une fermeture éclair, l'ensemble appelée *bootstrap*, pour aider à tirer les bottes. L'utilisation du terme *bootstrap* découle de la phrase "*pull oneself up by one's bootstrap*", attribuée à une histoire dans « *The Surprising Adventures of Baron Munchausen* » écrite par RUDOLF ERICH RASPE au XVIIIe siècle. BARON était tombé au fond d'un marécage, et quand il semblait que tout était perdu, il pensait prévoir s'en tirer avec ses propres *bootstrap*.

En économie, le *bootstrap* repose sur une méthode de simulation basée sur des données pour l'inférence statistique. Il a été proposé par EFRON (1992) comme une technique informatique afin d'estimer la précision d'un estimateur générique θ . En fait, le *bootstrap* est une méthode d'estimation du paramètre et des distributions à l'aide d'un ordinateur en rééchantillonnant les données d'origine et en calculant les statistiques à partir de données de simulation. L'avantage de cette méthode est d'être non-paramétrique. Ainsi, l'échantillon ne doit pas forcément suivre une distribution particulière. Mais, l'inconvénient est la complexité de calcul : il faut sans arrêt remettre l'échantillonnage. En somme, c'est une méthode non paramétrique utilisant une puissance de calcul très forte pour résoudre le problème de la répartition du poids statistique. Dans chaque échantillonnage, chaque échantillon peut être sélectionné seulement une fois, plusieurs fois ou ne jamais être sélectionné. Le nombre de sélections suit la loi binomiale, $Bin(n, 1/n)$. La distribution conjointe totale suit la loi multinomiale, $MultiNomial(n, 1/n, \dots, 1/n)$. Par conséquent, le *bootstrap* est une méthode permettant de générer la loi multinomiale.

La définition du *Moving Block Bootstrap* Puisque le *bootstrap* est une solution violente, lorsque la quantité de données est assez élevée, il devient très compliqué à calculer.

Par exemple, il ne peut être calculé en parallèle pour générer la loi multinomiale précédente. De plus, dans le flux de données (streaming), nous ne pouvons connaître la taille des données (n) à l'avance. SINGH (1981) a indiqué, dans la thèse d'EFRON, que les données utilisées étaient indépendantes et identiquement distribuées, mais que celles des séries chronologiques avaient une certaine structure de dépendance. Dans ce-cas là, si l'on continue à utiliser le *bootstrap* standard, on aboutit à l'échec. COGNEAU et ZAKAMOU-LINE (2010) se demandent s'il existe une dépendance en série dans les données ; pour cela il faut utiliser le *Moving Block Bootstrap*. Très facile à utiliser en mathématiques, il peut aussi fonctionner en parallèle. Le *Moving Blok Bootstrap* résout le problème des grandes données (n est grand) et des flux de données (n est incertain). La distribution simulée par le *Moving Block Bootstrap* est plus proche de la vraie distribution par rapport à la méthode *bootstrap* standard. HÄRDLE, HOROWITZ et KREISS (2003) ont prouvé que la précision de l'estimation des paramètres des données dépendantes en série fournies par la méthode *Moving Block Bootstrap* était meilleur que celle de l'estimation des paramètres des données dépendantes en série fournies par la méthode *bootstrap* standard en 2003. VOGEL et SHALLCROSS (1996) ont montré que le *Moving Block Bootstrap* pouvait constituer une alternative simple et attrayante pour les modèles ARMA multivariés plus complexes.

Le *Moving Block Bootstrap* est introduit en premier par KUNSCH (1989). Imaginons que nous avons une série de l'échantillon aléatoire X_1, X_2, \dots, X_n , si $B_i = \{X_i, \dots, X_{i+l-1}\}$ est un i ème bloc de la série de temps, alors le nombre des blocs dans le *bootstrap* $N=n-l+1$. Pour X_i , $1 \leq i \leq N$, l signifie que la longueur du même bloc et aussi le nombre des agents de ce bloc dans le modèle *basé agents*, n signifiant le nombre total des agents. Car dans le *Moving Block Bootstrap*, les blocs se chevauchent. Regardons la figure 3.10 :

FIGURE 3.10 : Les 6 agents se chevauchent dans le *Moving Block Bootstrap*



Dans cette figure, il y a 6 agents. Imaginons que chaque fois l'on place 3 agents dans un bloc ; dans ce cas-là, le $N = 6 - 3 + 1 = 4$. Il existe 4 blocs. $B_1 = \{7, 2, 9\}$, $B_2 = \{2, 9, 8\}$, $B_3 = \{9, 8, 6\}$, $B_4 = \{8, 6, 3\}$. Les agents 2, 9, 8, 6 sont sélectionnés plusieurs fois dans des blocs différents.

Afin de former un échantillon en utilisant le *Moving Block Bootstrap*, il faut choisir au hasard un certain nombre de blocs de l'ensemble $\{B_1, \dots, B_N\}$. Par conséquent, B_i^*, \dots, B_k^* représentent un échantillon aléatoire avec des répétitions de l'ensemble $\{B_1, \dots, B_N\}$, où chaque bloc contient le même nombre d'agents. On peut définir le paramètre de l'estimation par le *Moving Block Bootstrap* $\theta_{m,n} = T(F_{m,n})$, où $F_{m,n}$ montre la distribution empirique des échantillons *Moving Bloc Bootstrap* X_1, \dots, X_m .

En tant que blocs rééchantillonnés des agents $\{X_1, \dots, X_l\}, \{X_2, \dots, X_{l+1}\}, \dots, \{X_{(k-1)l+1}, \dots, X_{kl}\}$, ils sont indépendants et identiquement distribués avec une même probabilité conditionnelle $P = N^{-1}$. Quand $l \rightarrow \infty$, toute distribution de probabilité conjointe de dimension finie peut être révélée éventuellement à partir des observations rééchantillonnées. En conséquence, le *Moving Block Bootstrap* est capable d'approfondir efficacement les caractéristiques du processus dans toute la population rééchantillonnée.

Le choix de la longueur du bloc Les estimations fournies par la méthode *Moving Block Bootstrap* sont généralement biaisées et plutôt imprécises. En particulier, l'utilisation de blocs superposés entraîne un biais d'estimation potentiellement sérieux. Par conséquent, les estimations obtenues par la méthode *Moving Block Bootstrap* doivent être ajustées pour le biais. Une méthode d'ajustement de biais plus élaborée repose sur le choix d'une longueur de bloc appropriée dans la méthode *Moving Block Bootstrap*.

HALL, HOROWITZ et JING (1995) ont abordé la question du choix optimal des blocs dans les applications de la méthode *Moving Block Bootstrap* aux données dépendantes. Ils ont trouvé que la taille optimale du bloc dépendait significativement du contexte,

soit égale à $n^{1/3}$, $n^{1/4}$, $n^{1/5}$, dans les cas de variance ou d'estimation de biais, l'estimation d'une fonction de distribution unilatérale et l'estimation d'une fonction de distribution bilatérale. BÜHLMANN et KÜNSCH (1999) ont proposé une méthode entièrement axée sur les données pour sélectionner cette longueur de bloc. Cette méthode est basée sur l'équivalence de la variance en groupe du *bootstrap* à un estimateur de poids de la densité spectrale à l'origine. Dans cette équivalence, la longueur du bloc est l'inverse de la bande passante. Nous allons comparer la méthode de simulation ordinaire et la méthode *Moving Block Bootstrap* pour simuler ρ_1 par un processus *AR(1)*, afin de savoir quelle longueur de bloc est plus proche de celle dans la réalité sous le compilateur C++.

Avant de commencer l'estimation, nous devons définir les paramètres initiaux.

```
mt19937  rng  ;
normal_distribution<float>  random_normal  ;
constexpr  size_t  T = 1000  ;
constexpr  double  rho = .5  ;
constexpr  double  mu = 1  ;
```

Nous choisissons un générateur de nombres pseudo-aléatoires *Mersenne Twister* qui est le plus rapide dans l'ensemble du générateur de nombres pseudo-aléatoires afin de produire 1000 échantillons qui suivent la loi normale. Nous choisissons un coefficient d'auto-corrélation $\rho = 0,5$ et le terme constant $\mu = 1$. Le coefficient d'auto-corrélation doit être supérieur à 0,5, sinon les prévisions sont inexactes.

Estimation de la variance de l'estimateur de ρ_1 pour un *AR(1)* par simulation

D'abord, nous avons défini le nombre d'estimations, Réplication = 100 et l'estimateur ρ_1 . En C++, nous devons taper `constexpr size_t R = 100000` et `float rho_1[R]`. Le processus d'un *AR(1)* est

$$X_t = \mu + \rho * X_{t-1} + \epsilon_t \quad (3.7)$$

ϵ_t représente un bruit blanc ; il est indépendant et identiquement distribué, avec une moyenne nulle et une variance σ^2 . Si la moyenne est définie en x , alors :

$$E(X_t) = E(\mu) + \rho * E(X_{t-1}) + E(\epsilon_t) \quad (3.8)$$

$$x = \mu + \rho * x + 0 \quad (3.9)$$

$$x = \frac{\mu}{1 - \rho} \quad (3.10)$$

En C++, nous avons réalisé les programmes suivants :

```
double x[T] ;
double x_m1 = mu / (1. - rho) ;
for ( size_t = 0 ; t < T ; ++t ) {
const auto xx = rho*x_m1 + mu + random_normal(rng) ;
x[t] = xx ;
x_m1 = xx ;}
```

Ensuite, pour chaque estimation, nous calculons la moyenne de x , \bar{X} , puis tous les ρ_1 par l'équation suivante :

```
const auto xb = accumulate(x, x+T, 0.) / T ;
for_each(x, x+T, [=](double & x) { x -= xb ; }) ;
rho_1[r] = inner_product(x+1, x+T, x, 0.) /
inner_product(x, x+T, x, 0.) ;
```

À la fin, nous calculons la moyenne de ρ_1 et sa variance.

```
const auto m = accumulate(rho_1, rho_1+R, 0.) / R ;
const auto s2 = inner_product(rho_1, rho_1+R, rho_1,
0.) / R ;
```

TABLE 3.5 : Résultat de l'estimation par simulation

ρ_1	variance
0.497	0.00075

Estimation de la variance de l'estimateur ρ_1 pour un $AR(1)$ par le *Moving Block Bootstrap* Pour obtenir les résultats avec différentes longueurs de blocs, nous mettons `size_t number = 2; number <= 10; ++ number`. Pour sélectionner un agent par la méthode de *moving block bootstrap*, nous implantons un *vector*, nous l'appelons `one_draw_from_moving_block_bootstrap` et la sélection d'agent est la suivante :

```
auto xx = one_draw_from_moving_block_bootstrap(rng, x,
        T/number) ;
```

Dans ce *vector*, nous utilisons aussi un générateur de nombres pseudo-aléatoires *Mersenne Twister mt19937 & rng* pour fabriquer des agents `vector <double> x`, et définissons la taille de blocs `size_t block_size`. Pour être sûr que le programme fonctionne, il faut nous assurer que la taille des agents soit supérieure à celle du bloc `assert (x.size() > block_size)`. Ensuite, nous devons calculer le nombre de blocs.

```
const size_t blocks_nr = static_cast<double>(x.size() /
        block_size + .5) ;
```

Nous utilisons la taille des agents divisée par la longueur du bloc pour obtenir le nombre de blocs. Il faut ajouter 0,5 à la fin pour s'assurer que le dernier bloc est choisi. Cela évite le problème de chiffres arrondis. Si le dernier bloc ne comprend pas assez d'agents, s'il n'égalent pas la longueur du bloc, il faut remplir les places restant dans ce bloc.

```
uniform_int_distribution<size_t> index(0, x.size()-
        block_size) ;
vector<double> tmp ; tmp.reserve(blocks_nr*block_size)
;
```

```

for ( size_t i = 0 ; i < blocks_nr ; ++ i ) {
    const auto start = index(rng) ;
    copy(x.begin()+start , x.begin()+start+
        block_size , back_inserter(tmp)) ;
}

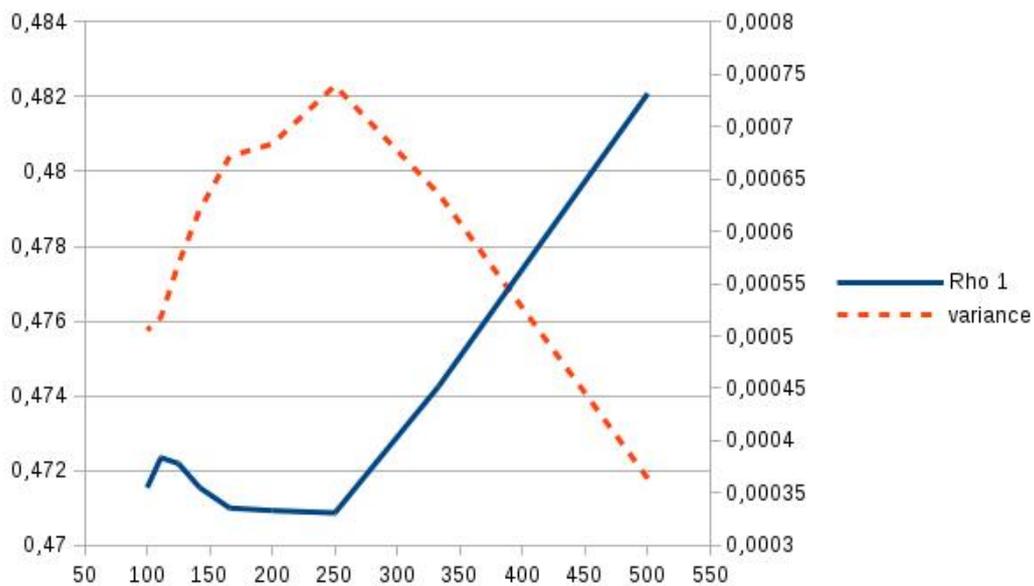
```

La table 3.6 montre le résultat de l'estimation par le *moving block bootstrap*, $T=1000$.

TABLE 3.6 : Résultat de l'estimation par le *moving block bootstrap*

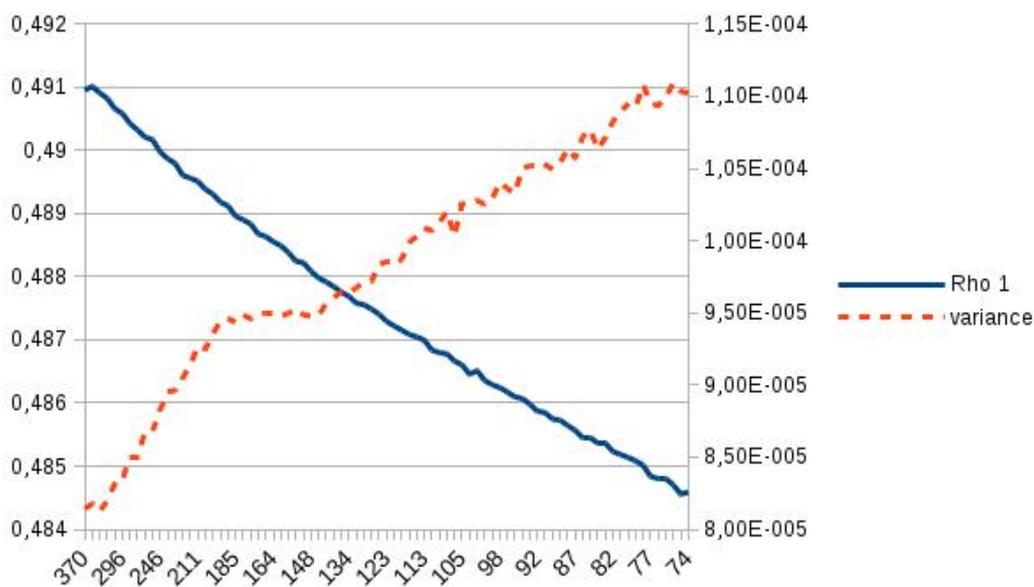
nombre de bloc	longueur de bloc	ρ_1	variance
2	500	0.482	0.00036
3	333	0.474	0.00064
4	250	0.471	0.00074
5	200	0.471	0.00068
6	166	0.471	0.00067
7	142	0.472	0.00062
8	125	0.472	0.00057
9	111	0.472	0.00052
10	100	0.472	0.00051

FIGURE 3.11 : ρ_1 et sa variance estimée par le *moving block bootstrap*, $T=1000$, $R=100000$



La figure 3.12 montre le résultat estimé dont le nombre des échantillons est de 7400 et la réplication est de 100 000. Par rapport à la figure 3.11, on voit clairement que la meilleure longueur du bloc change avec le changement de taille des échantillons. Pour estimer la fonction objective, nous prenons le résultat dont la taille des échantillons est plus proche des données historiques que nous avons choisies. $T = 7400$. Selon la figure 3.12, la meilleure longueur du bloc est d'environ 130. Ainsi, pour calculer pratiquement, nous prenons 150 comme longueur du bloc dans notre modèle. Cela veut dire que dans chaque bloc, il y a des données environ pendant 6 mois.

FIGURE 3.12 : ρ_1 et sa variance estimée par le *moving block bootstrap*, $T=7400$, $R=100000$



3.5.2 La fonction objective $f(\theta)$ pour la simulation

Sur la base de l'identification des statistiques robustes des séries chronologiques des rendements des actions, une fonction objective est dérivée. Cette fonction prend en compte les faits stylisés sur la distribution inconditionnelle des rendements des actions et des propriétés de la distribution conditionnelle, notamment, l'hétéroscédasticité conditionnelle auto-régressive (*ARCH*) et la mémoire longue. Nous pouvons obtenir une estimation

de la matrice de variance-covariance des différents moments inclus dans la fonction de l'objectif par la méthode *moving block bootstrap* avec la longueur 150, qui sert de base à la matrice de pondération.

La méthode des moments simulés Dans notre modèle, nous utilisons un outil d'évaluation des modèles *basés agents* reposant sur l'inférence indirecte simulée, proposé par FABRETTI (2013). Cet outil est basé sur la comparaison des moments des données réelles et des moments simulés du modèle *basé agents*. Cela implique la minimisation d'une approximation stochastique d'une fonction objective :

$$\min_{\theta \in \Theta} f(\theta) \quad (3.11)$$

où θ est un réglage de paramètre fixe, Θ est un sous-espace de \mathbb{R}^d décrivant les valeurs des paramètres possibles. $f(\theta)$ mesure la distance entre les moments empiriques des données réelles et celle des séries chronologiques simulées. Les paramètres θ du modèle multi-agents ne sont pas considérés comme donnés, étant soumis à l'estimation. Une méthode standard du moment conditionnel est nécessaire $E(m^s | \theta) = m$. Nous supposons que $m = (m_1, m_2, \dots, m_k)$ correspondant à k moments du processus de rendement des actions pour correspondre au modèle *basé agents*. Bien que le vrai vecteur des moments m ne soit pas connu, il doit être estimé sur la base des observations disponibles \mathbb{R} . Donc $m^e = (m_1^e, m_2^e, \dots, m_k^e)$ représente des estimations constantes de m . Par conséquent, si m est donné, le vecteur de paramètre θ peut être estimé en remplaçant la valeur attendue par la moyenne sur les moments simulés. Dans ce cas, l'estimateur de moments simulés θ est la solution à

$$\frac{1}{I} \sum_{i=1}^I [(m_i^s | \theta) - m] = 0 \quad (3.12)$$

Si le nombre de moments conditionnels k dépasse celui des paramètres dans θ , une méthode de temps généralisée est indiquée. Nous mettons le vecteur $k * 1$ des écarts par

rapport au moment conditionnel ci-dessus comme :

$$G_I(\theta) = \sum_{i=1}^I [(m_i^s | \theta) - m] \quad (3.13)$$

Alors, la minimisation de la somme pondérée des carrés est

$$f(\theta) = \frac{1}{I} G_I^T W G_I \quad (3.14)$$

où W est une matrice définie positive $k \times k$, entraînant la méthode généralisée d'estimateur de moments simulés θ . La matrice de pondération W doit permettre des erreurs plus grandes pour les paramètres estimés avec un degré d'incertitude plus élevé. Le choix de W comme matrice de variance-covariance de $G_I(\theta)$ joue un rôle clé dans l'analyse des composantes principales. Cette analyse consiste à décomposer une matrice de covariance en valeurs singulières, afin de déterminer la direction des caractéristiques de la valeur singulière maximale. Au sens physique, cela consiste à calculer la corrélation entre les dimensions. Ainsi, concernant le résultat du calcul de la matrice de covariance, plus la valeur d'un élément est élevée, plus la corrélation entre les caractéristiques correspondantes de l'objet est élevée. G_I^T représente la transposition de G_I

La sélection des moments La sélection des moments doit prendre en compte l'objectif final de l'analyse, c'est-à-dire l'évaluation des modèles. D'après WINKER, GILLI et JELESKOVIC (2007), les statistiques doivent d'abord être robustes. Ensuite, elles doivent présenter le potentiel de discrimination entre les modèles alternatifs et les constellations de paramètres. Les statistiques sont mesurées par la méthode *moving block bootstrap*. Plus leur variance est faible, plus l'estimation est fiable. La variance de l'attente du moment estimé à partir des données simulées doit être faible afin de pouvoir discriminer différentes constellations des paramètres pour le même modèle. Ayant gardé ces critères à l'esprit, les statistiques suivants ont été sélectionnées.

TABLE 3.7 : Statistiques sélectionnées

étiquette	statistique	valeur du CAC 40	valeur estimée par le <i>mbb</i>
m_1	moyenne	0.00022	0.00021
m_2	sigma	0.014	0.0014
m_3	kurtosis	4.58	4.37
m_4	studentized range	14.63	13.86
m_5	ρ_1	-0.0022	-0.0050
m_6	ρ_{10}	-0.0012	-0.0002
m_7	ν_1	0.17	0.16
m_8	ν_{10}	0.20	0.18
m_9	χ_1	0.18	0.17
m_{10}	χ_{10}	0.21	0.18

La table 3.7 fournit un résumé des statistiques utilisées dans la fonction objective, y compris les détails de la spécification et les valeurs pour les séries chronologiques de rendement du *CAC 40* pour la période entre le 31/12/1987 et le 31/03/2017.

La moyenne, le sigma, le kurtosis et le *studentized range* sont sélectionnés pour représenter la forme globale de la distribution des données. Le ρ_1 , le ρ_{10} , le ν_1 , le ν_{10} , le χ_1 , le χ_{10} représentent la volatilité clustering par le rendement r_t , le rendement absolu $|r_t|$ et le rendement carré r_t^2 .

La pondération Dans notre programme, nous utilisons une bibliothèque d'analyses numériques en C++ *Eigen* comportant des outils d'algèbre linéaire, des opérations matricielles et vectorielles, des transformations géométriques pour faciliter le calcul de la matrice W . Nous mettons les 10 paramètres estimés dans une matrice :

```
Eigen :: MatrixXd X(R,K)
```

Nous donnons à chaque statistique une position dans la matrice :

```
X (r,0) = average ;
X (r,1) = sigma ;
X (r,2) = kurtosis ;
X (r,3) = studentized_range ;
```

```

X (r,4) = rho_1 ;
X (r,5) = rho_{10} ;
X (r,6) = nu_1 ;
X (r,7) = nu_{10} ;
X (r,8) = chi_1 ;
X (r,9) = chi_{10} ;

```

Nous utilisons `const Eigen::VectorXd average = X.colwise().mean()` pour obtenir des statistiques estimées par le *moving block bootstrap*. Puis nous calculons la valeur **W** et sa corrélation par les commandes suivantes :

```

const Eigen::MatrixXd Xtilde = X.rowwise() - average.transpose
() ;
const Eigen::MatrixXd Omega = (Xtilde.transpose() * Xtilde) / (
R-1) ;
const Eigen::VectorXd d = Omega.diagonal() ;
const Eigen::MatrixXd D = (d * d.transpose()).array().sqrt() ;
const Eigen::MatrixXd Corr = Omega.array() / D.array() ;

```

où **Omega** est le **W**. Une estimation $\hat{\Sigma}_{\text{BB}}$ de la matrice de variance-covariance $\text{Var}(\mathbf{m}^e)$ est obtenue à partir de la distribution de **m** estimée par la méthode de *moving block bootstrap*. Ici, la longueur du bloc $L = 150$ et la réplication $R = 100\,000$.

L'estimation d'une matrice de pondération optimale **W** est compliquée car $G_1(\theta)$ est soumis à deux types d'incertitude résultant de l'estimation des moments empiriques pour appairer **m** et des moments simulés pour un vecteur de paramètre donné $E(m^s|\theta)$. Alors que l'incertitude concernant les moments simulés peut être réduite en augmentant le nombre de simulations, l'incertitude dans l'estimation de **m** doit être prise en compte.

Dans ce cas-là, WINKER propose d'utiliser $\mathbf{W} = \text{Var}^{-1}(\mathbf{m}^e)$. Alors

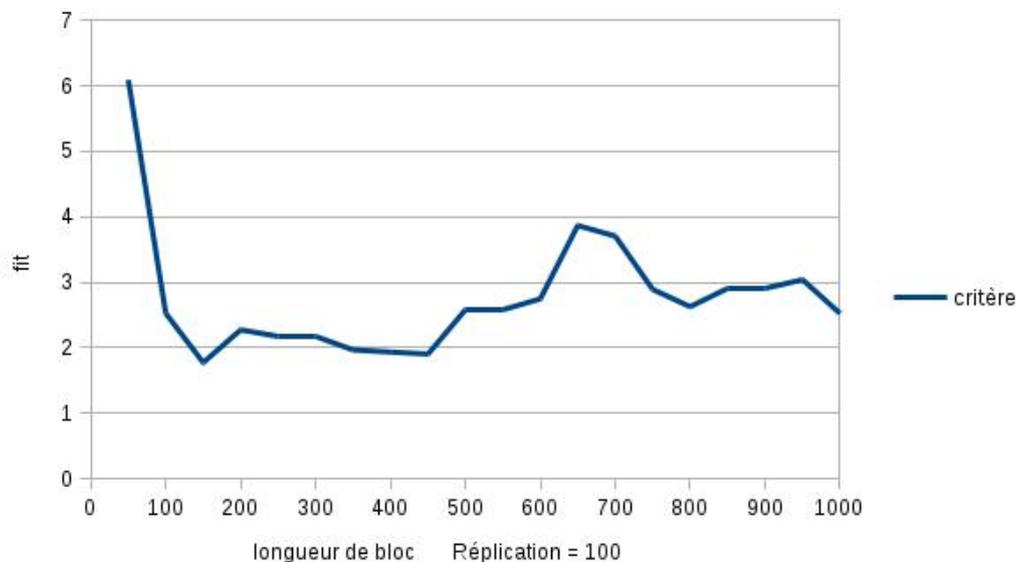
$$\widehat{\mathbf{W}}(\mathbf{m}^e) = \sum_{\text{BB}}^{-1} \quad (3.15)$$

Ainsi, notre fonction objective pour l'évaluation d'un vecteur de paramètre θ basé sur les données simulées est donnée ci-dessous :

$$f(\theta) = \frac{1}{\mathbf{I}} \widehat{\mathbf{G}}_1' \widehat{\mathbf{W}}(\mathbf{m}^e) \widehat{\mathbf{G}}_1' \quad (3.16)$$

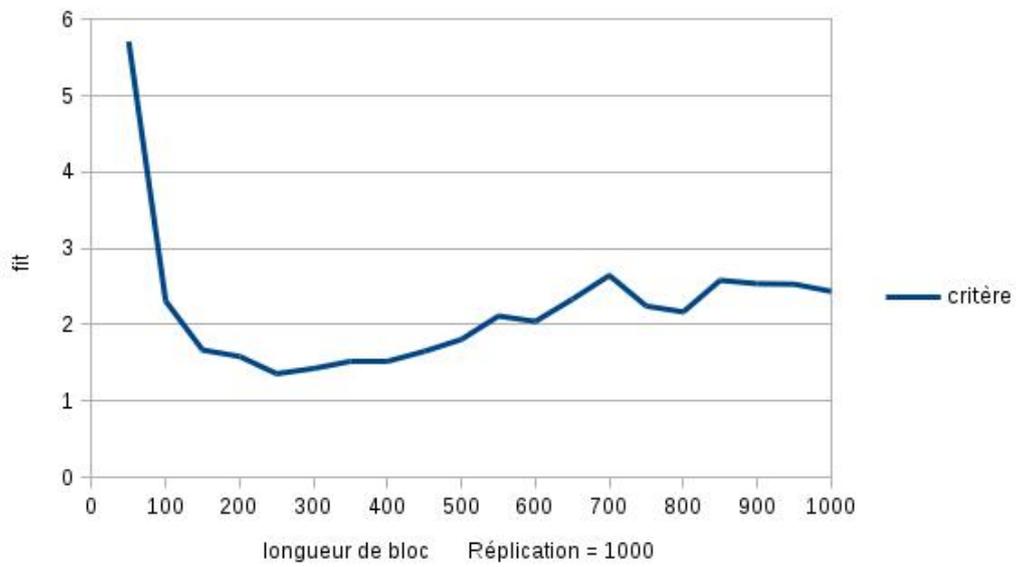
$$= \left\{ \frac{1}{\mathbf{I}} \sum_{i=1}^{\mathbf{I}} 1[(m_i^s|\theta) - m^e] \right\}' \widehat{\mathbf{W}}(\mathbf{m}^e) \left\{ \frac{1}{\mathbf{I}} \sum_{i=1}^{\mathbf{I}} 1[(m_i^s|\theta) - m^e] \right\} \quad (3.17)$$

FIGURE 3.13 : Changement de la valeur de fit à mesure des différentes longueurs de bloc, R=100



Le critère Les figures 3.13, 3.14, 3.15 et 3.16 montrent que la variation de la valeur du critère selon différentes longueurs du bloc, sous les nombres d'estimations 100, 1000, 10000 et 100000. On voit clairement que, quand $L = 250$, on peut obtenir le critère le plus petit. La table 3.8 montre le détail de l'estimation avec une réplification différente. Vu le résultat des figures 3.15 et 3.16, nous constatons qu'après la réplification 10000,

FIGURE 3.14 : Changement de la valeur de fit à mesure des différentes longueurs de bloc, R=1000



les valeurs de fit deviennent plus stables.

FIGURE 3.15 : Changement de la valeur de fit à mesure des différentes longueurs de bloc, R=10000

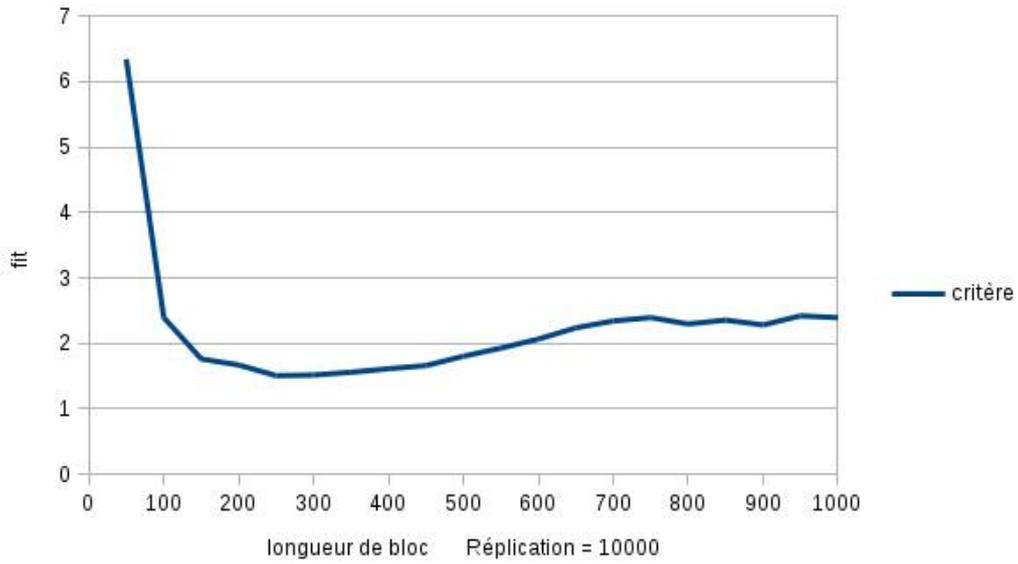
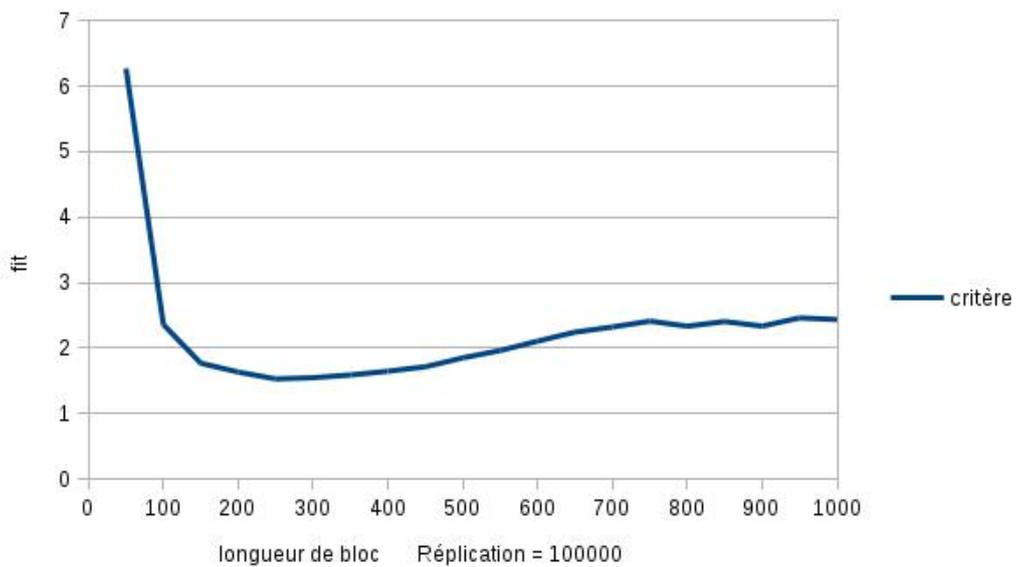


FIGURE 3.16 : Changement de la valeur de fit à mesure des différentes longueurs de bloc, R=100000



$$\mathbf{W} = \begin{Bmatrix}
 2.27\text{E}-08 & -7.21\text{E}-08 & -3.76\text{E}-05 & 1.87\text{E}-05 & 5.27\text{E}-07 & 2.59\text{E}-07 & -1.53\text{E}-06 & -2.19\text{E}-06 & -7.69\text{E}-07 & -2.39\text{E}-06 \\
 -7.21\text{E}-08 & 6.59\text{E}-07 & 0.0003 & -0.0002 & -2.10\text{E}-06 & -1.60\text{E}-06 & 1.16\text{E}-05 & 1.84\text{E}-05 & 7.78\text{E}-06 & 2.35\text{E}-05 \\
 -3.76\text{E}-05 & 0.0003 & 1.3576 & 0.7480 & -0.0086 & 0.0025 & 0.0217 & 0.0297 & 0.0200 & 0.0397 \\
 1.87\text{E}-05 & -0.0002 & 0.7480 & 1.2371 & -0.0048 & 0.0033 & 0.0067 & 0.0070 & 0.0119 & 0.0146 \\
 5.27\text{E}-07 & -2.10\text{E}-06 & -0.0086 & -0.0048 & 0.0002 & 3.53\text{E}-05 & -8.96\text{E}-05 & -0.0002 & -0.0001 & -0.0002 \\
 2.59\text{E}-07 & -1.60\text{E}-06 & 0.0025 & 0.0033 & 3.53\text{E}-05 & 0.0002 & 3.65\text{E}-05 & -4.50\text{E}-05 & 5.83\text{E}-05 & -3.08\text{E}-05 \\
 -1.53\text{E}-06 & 1.16\text{E}-05 & 0.0217 & 0.0067 & -8.96\text{E}-05 & 3.65\text{E}-05 & 0.0007 & 0.0008 & 0.0005 & 0.0009 \\
 -2.19\text{E}-06 & 1.84\text{E}-05 & 0.0297 & 0.0070 & -0.0002 & -4.50\text{E}-05 & 0.0008 & 0.0014 & 0.0006 & 0.0017 \\
 -7.69\text{E}-07 & 7.78\text{E}-06 & 0.0200 & 0.0119 & -0.0001 & 5.83\text{E}-05 & 0.0005 & 0.0006 & 0.0007 & 0.0008 \\
 -2.39\text{E}-06 & 2.35\text{E}-05 & 0.0397 & 0.0146 & -0.0002 & -3.08\text{E}-05 & 0.0009 & 0.0017 & 0.0008 & 0.0025
 \end{Bmatrix}$$

$$\begin{array}{c}
\mathfrak{g} \\
\text{Correlation} =
\end{array}
\left\{
\begin{array}{cccccccc}
1 & -0.59018 & -0.21417 & 0.11151 & 0.24141 & 0.11051 & -0.39643 & -0.39251 & -0.19956 & -0.31994 \\
-0.59018 & 1 & 0.35705 & -0.18707 & -0.17837 & -0.12646 & 0.55969 & 0.61281 & 0.37477 & 0.58136 \\
-0.21417 & 0.35705 & 1 & 0.57722 & -0.50660 & 0.13972 & 0.72923 & 0.68779 & 0.67145 & 0.68613 \\
0.11151 & -0.18707 & 0.57722 & 1 & -0.29753 & 0.18812 & 0.23622 & 0.17021 & 0.41964 & 0.26396 \\
0.24141 & -0.17837 & -0.50660 & -0.29753 & 1 & 0.15633 & -0.24159 & -0.29507 & -0.30891 & -0.31338 \\
0.11051 & -0.12646 & 0.13972 & 0.18812 & 0.15633 & 1 & 0.09154 & -0.07795 & 0.14625 & -0.03974 \\
-0.39643 & 0.55969 & 0.72923 & 0.23622 & -0.24159 & 0.09154 & 1 & 0.79994 & 0.79303 & 0.69648 \\
-0.39251 & 0.61281 & 0.68779 & 0.17021 & -0.29507 & -0.07795 & 0.79994 & 1 & 0.59762 & 0.92331 \\
-0.19956 & 0.37477 & 0.67145 & 0.41964 & -0.30891 & 0.14625 & 0.79303 & 0.59762 & 1 & 0.60820 \\
-0.31994 & 0.58136 & 0.68613 & 0.26396 & -0.31338 & -0.03974 & 0.69648 & 0.92331 & 0.60820 & 1
\end{array}
\right.$$

$$\mathbf{W}^1 = \begin{Bmatrix}
7.41\text{E} + 007 & 6.84\text{E} + 006 & -6126.72 & 2843.38 & -354989 & 7850.43 & 251234 & 78615.4 & -108903 & -29483.1 \\
6.84\text{E} + 006 & 4.04\text{E} + 006 & -425.675 & 1503.16 & 19506.6 & -8283.43 & -1843.01 & 4177.69 & -27651.1 & -22730.3 \\
-6126.72 & -425.675 & 4.41618 & -1.60966 & 121.637 & -25.9019 & -93.2522 & -24.4225 & 30.9041 & -9.81262 \\
2843.38 & 1503.16 & -1.60966 & 2.43912 & 10.3063 & -7.02024 & 41.4231 & 1.55015 & -38.1516 & -2.61208 \\
-354989 & 19506.6 & 121.637 & 10.3063 & 12295.5 & -2069.32 & -2922.16 & -2401.9 & 1319.73 & 437.153 \\
7850.43 & -8283.43 & -25.9019 & -7.02024 & -2069.32 & 5752.42 & -705.964 & 3377.44 & -517.74 & -1319.22 \\
251234 & -1843.01 & -93.2522 & 41.4231 & -2922.16 & -705.964 & 10705.1 & -5326.48 & -6079.27 & 2750.65 \\
78615.4 & 4177.69 & -24.4225 & 1.55015 & -2401.9 & 3377.44 & -5326.48 & 11917.4 & 2587.85 & -6445.16 \\
-108903 & -27651.1 & 30.9041 & -38.1516 & 1319.73 & -517.74 & -6079.27 & 2587.85 & 6523.68 & -1681.47 \\
-29483.1 & -22730.3 & -9.81262 & -2.61208 & 437.153 & -1319.22 & 2750.65 & -6445.16 & -1681.47 & 4514.37
\end{Bmatrix}$$

TABLE 3.8 : La comparaison de différents nombres de simulation

longueur de bloc	fit, R = 100	fit, R = 1000	fit, R = 10000	fit, R = 100000
50	6.09	5.71	6.34	6.27
100	2.52	2.31	2.39	2.36
150	1.77	1.67	1.76	1.77
200	2.27	1.58	1.67	1.63
250	2.17	1.36	1.50	1.53
300	2.17	1.42	1.51	1.55
350	1.97	1.52	1.56	1.59
400	1.93	1.52	1.61	1.65
450	1.90	1.65	1.66	1.71
500	2.58	1.81	1.80	1.85
550	2.59	2.11	1.93	1.96
600	2.75	2.04	2.06	2.10
650	3.87	2.33	2.23	2.24
700	3.71	2.64	2.34	2.32
750	2.89	2.24	2.39	2.41
800	2.63	2.17	2.29	2.33
850	2.90	2.58	2.35	2.40
900	2.90	2.54	2.28	2.33
950	3.04	2.53	2.42	2.46
1000	2.52	2.44	2.39	2.43

Chapitre 4

Notre propre modèle *basé agents* et nos résultats

Dans le Chapitre 3, nous avons prouvé que le rendement du marché boursier français, le *CAC 40*, avait des faits stylisés : la queue épaisse, la mémoire longue et la volatilité clustering. Mais ce résultat ne peut pas refléter la distribution de probabilité des chocs exogènes qui frappent le marché. On ne peut savoir si les phénomènes d'anomalies sont liés aux interactions des traders et au processus du trading des actions.

Dans le marché boursier, beaucoup de traders suivent l'analyse technique. Ils pensent que toutes les fluctuations des prix des actions sont affichées d'après des indicateurs techniques. Cette idée est contraire à l'ordre du marché, car la volatilité des prix sur le marché est la vraie raison, l'affichage des indicateurs techniques étant le résultat. Si l'on considère les indicateurs techniques comme une chose sacro-sainte, et que l'on ignore l'essence de la volatilité du marché lui-même, quand le marché boursier montre ses caractéristiques d'origine, la méthode d'analyse technique subit également une défaite écrasante. Voilà pourquoi de nombreux experts, qui ont utilisé des indicateurs techniques dans le trading des actions, ont été saints et saufs durant de nombreuses années, mais

ils ont été soudain éliminés pendant la grande volatilité du marché.

Ainsi, le vrai grand maître du commerce des actions devrait être un suiveur du marché boursier. Il n'est jamais superstitieux face aux indicateurs techniques et aux analyses fondamentales, mais il considère la volatilité des prix comme le seul standard à juger et à opérer. Car le marché n'a jamais tort, c'est chaque trader qui se croit le détenteur de la vérité qui a tort. En fait, un facteur passe inaperçu : c'est le comportement collectif et la psychologie des investisseurs sur le marché boursier. Toutes les variations de prix des actions du marché financier ne sont que l'anticipation des prix et la réfraction psychologique de la majorité des traders. Comme SOROS (1994) l'a indiqué dans sa théorie de la réflexivité. Cette réflexivité signifie l'interaction entre les investisseurs et le marché. La théorie de la réflexivité fait référence à l'interaction entre les investisseurs et les marchés financiers. Les premiers forment une anticipation du marché et agissent selon les informations acquises et leur connaissance du marché. Cette action change la direction initiale du développement du marché en reflétant une forme nouvelle, en donnant de nouvelles informations qui font que les investisseurs produisent de nouvelles idées d'investissement et continuent de changer la direction du marché financier. Par exemple, dans un marché boursier, lorsque tous les investisseurs ou la plupart estiment que le cours d'une action va augmenter (baisser), ils déterminent leur comportement commercial selon cette estimation.

On considère que le marché financier est vulnérable face à un comportement collectif réagissant par un grand groupe d'agents qui placent le même ordre simultanément. BANERJEE (1992) a analysé un modèle de décision séquentielle dans lequel chaque agent examine les décisions prises par les agents précédents, quand il prend sa décision. Ce phénomène est appelé le comportement grégaire. Comment modéliser la façon dont les décisions des agents sont influencées par des interactions mutuelles ? Dans ce chapitre, nous allons montrer comment construire un tel modèle *basé agents* en utilisant le compilateur C++ pour mimétiser la dynamique du prix du marché boursier français CAC

40. Nous montrons qu'une disposition d'objets orientée pour concevoir les attributs des individus ne conduit pas à des accès efficaces à la mémoire. Nous proposons aussi un algorithme d'échanges d'opinions dans le voisinage spatial sans besoin d'exclusion mutuelle. Cet algorithme est basé sur des idées de la théorie de la percolation que nous avons introduite dans le premier Chapitre. Les agents occupent le nœud d'un réseau et sont influencés par les décisions de leurs voisins les plus proches. Ils sont hétérogènes, garantissant qu'ils ne prennent pas spontanément les mêmes décisions. Nous pensons que ce modèle *basé agents* peut expliquer certains des faits stylisés des rendements boursiers français *CAC 40* que nous avons mentionnés dans le Chapitre 3. L'élément clé de notre recherche n'est pas la description exacte du comportement individuel, mais l'interaction entre les agents hétérogènes et le mécanisme de rétroaction sur les opérations futures et la génération de la volatilité clustering.

4.1 Notre modèle *basé agents*

4.1.1 Agents positionnés dans un réseau

Dans notre modèle, nous considérons que tous les agents sont positionnés dans un réseau. Ce réseau est une grille carrée $L * L$. Chaque agent occupant une cellule est connecté avec ses quatre voisins les plus proches, le voisin de l'Est, le voisin de l'Ouest, le voisin du Nord et le voisin du Sud. Ainsi, nous voulons donner à chaque agent des coordonnées; nous considérons donc cette grille carrée comme une matrice en mathématiques. Ensuite, nous expliquerons comment transformer leurs coordonnées mathématiques en informatique sous le compilateur C++. Dans ce compilateur, quand on fait le calcul, tous les agents sont positionnés sur une ligne. Chacun a un numéro, on l'appelle le numéro d'agent. Ce numéro commence par 0 jusqu'au $L * L - 1$, le nombre total des agents moins un. Concernant sa position dans la matrice (j,i) , $j = n/L$, $i = n - j * L$, où j est égal à la partie

entière de n divisé par L . D'abord, nous devons calculer les numéros de ses quatre voisins les plus proches. Cela est important pour le processus suivant. Nous commençons par son le voisin de l'Est, parce que le soleil apparaît au-dessus de l'horizon, vers l'Est. Nous utilisons une commande pour calculer le numéro du voisin de l'Est : $return (i+1==L) ? j*L : (i+1)+j*L$. En informatique, cette commande est expliquée comme suit : $?$ signifie que la valeur du retour (numéro de l'agent voisinage) dépend de l'équation $i+1==L$, si $i+1==L$ est correcte, la valeur du retour est égale à celle devant le colon $j*L$; si l'équation $i+1==L$ n'est pas correcte, la valeur du retour est égale à la valeur derrière le colon $(i+1)+j*L$. De même, nous définissons les numéros des trois autres voisins par les commandes suivantes : pour le voisin de Sud, $return (m>=L*L) ? m-L*L : m$, où $m = i + (j+1)*L$; pour le voisin de l'Ouest, $return (i==0) ? L-1 + j*L : i-1 + j*L$. Voici un exemple :

FIGURE 4.1 : Les 9 agents dans une ligne sous C++

0	1	2	3	4	5	6	7	8
A	B	C	D	E	F	G	H	I

Imagions que notre réseau comporte 9 agents. Ils sont positionnés dans une grille carrée dans la figure 4.2, $L = 3$. La figure 4.3 correspond aux coordonnées des agents dans la matrice, et la figure 4.1 montre leurs numéros et leurs positions en C++. À présent, nous choisissons un agent E , son numéro est 4. Selon les commandes pour calculer ses coordonnées, $j = 4/3$, on prend la partie entière, alors que $j = 1$, $i = n - j*L = 4 - 1*3 = 1$, ses coordonnées est (1,1). Le numéro de son voisin à l'Est F est $(1+1==3) ? 1*3 : (1+1) + 1*3$, d'après la règle de calcul en C++, nous obtenons le numéro 5 de F . De même, le numéro de son voisin au Sud H est 7, celui de son voisin à l'Ouest D est 3, celui de son voisin au Nord B est 2. Par ailleurs, il convient de noter que pour l'agent A , son voisin au Nord est G , et son voisin à l'Ouest est C . Cela signifie que les agents en haut du réseau sont liés à ceux du bas et les agents du côté gauche sont liés à ceux du côté droit. De cette façon, le réseau suppose la topologie d'un tore.

La transformation entre le calcul de la matrice et le calcul en C++ est importante. C'est la base de notre modèle pour simuler les phénomènes anomaux du CAC 40.

FIGURE 4.2 : Les 9 agents dans le réseau d'une grille carrée, $L = 3$

A	B	C
D	E	F
G	H	I

FIGURE 4.3 : Les coordonnées des 9 agents dans une matrice

(0,0)	(0,1)	(0,2)
(1,0)	(1,1)	(1,2)
(2,0)	(2,1)	(2,2)

4.1.2 Mécanisme de la prise de décision

À présent, tous les agents se trouvent dans leurs positions. Nous allons ici déterminer quelques règles de transaction et les situations initiales pour les agents. Nous donnons à chacun le même montant initial $vector<float> money(N,100)$ et la même quantité d'actions $vector<float> stock(N,100)$. Nous introduisons un choc idiosyncrasique $vector<float> shock(N)$ tombant au début de chaque période pour chaque agent. Chacun d'eux peut émettre une décision de commerce $vector<char> decision(N)$ au moment t : 1 l'achat d'une action, -1 la vente d'une action et 0 ni l'achat ni la vente d'une action, le maintien de la situation précédente. Mais nous mettons une limite pour le commerce des actions : chaque agent peut acheter ou vendre une seule action à chaque moment. Le $vector<char> previous_decision(N)$ est la décision de commerce précédente.

La prise d'une décision d'un agent est influencée par celles de ses 4 voisins les plus proches et le bruit idiosyncratique. Au début de chaque période de commerce, chaque agent reçoit un choc idiosyncratique $C_i(t)$ qui décrit les chocs de sa préférence person-

nelle et se maintient constant tout au long de la période. Ce choc idiosyncratique est un nombre aléatoire généré par le générateur de nombres aléatoires qui suit une distribution normale $generate(RANGE(shock), [&] return rand_normal(rng)$; La distribution de C_i est uniforme sur l'intervalle $[-1,1]$. De plus, chaque agent échange à plusieurs reprises des informations avec ses 4 voisins les plus proches $D(\bar{i})$, au cours de chaque période t . On note :

```
neighbor_decision =
    is_east_connected(L, connections, n) *
        decision[east(L,n)] +
    is_south_connected(L, connections, n) *
        decision[south(L,n)] +
    is_west_connected(L, connections, n) *
        decision[west(L,n)] +
    is_north_connected(L, connections, n) *
        decision[north(L,n)] ;
```

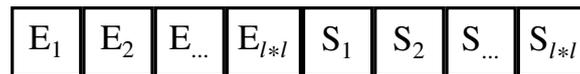
Dans le programme ci-dessus, $decision[east(L,n)]$ représente la prise de décision du voisin à l'Est de l'agent $_i$, L étant la taille de la matrice et n le numéro de l'agent. $is_east_connected(L, connections, n)$ représente la valeur du retour de la fonction *bool*, soit 1, soit 0. Nous la définissons comme suit en C++ :

```
static inline bool & is_east_connected (
    UInt L,
    const std::vector<char> & connections,
    UInt n) {
    return connections[n] }
```

Le *bool*, en programmation informatique est un type de variable à deux états. Les variables de ce type sont soit à l'état vrai soit à l'état faux. $is_east_connected$ est une forme de

question : si l'agent est connecté avec son voisin à l'Est, $return\ connections[s]$ représente la valeur retournant au côté Est de l'agent i . Comme dans le modèle de la percolation des informations, si l'agent i est connecté avec son voisin à l'Est, la valeur du côté de l'agent i est vraie, alors la valeur est 1 ; sinon, c'est 0. Nous avons expliqué la transformation entre le calcul de la matrice et le calcul en C++ dans les figures 4.1 et 4.2 ; imaginons que tous les agents sont positionnés dans une grille carrée avec $L = 3$, si nous ne considérons que les états du côté de l'Est et du Sud de tous les agents, nous pouvons décrire tous les états de connexion. Par exemple, si le côté de l'Est et le côté du Sud de chaque agent sont connectés avec leur voisin, on peut dire que tous les agents de cette grille $3*3$ sont connectés. Nous définissons $connections$ comme le côté d'un agent. Alors que $return\ connections(n)$ représente le côté de l'Est pour l'agent i , $return\ connections(L*L+n)$ représente celui du Sud pour l'agent i , $return\ connections[west(L,n)]$ représente celui de l'Ouest pour l'agent i , $return\ connections[L*L+north(L,n)]$ représente le côté du Nord pour l'agent i . Si le côté est connecté, la vraie valeur retourne à 1, cela signifie que les deux agents aux deux côtés sont connectés ; si le côté n'est pas connecté, la vraie valeur retourne à 0, cela indique que les deux agents aux deux côtés ne sont pas connectés. La figure 4.4 représente clairement cette relation.

FIGURE 4.4 : Tous les côtés de l'Est et du Sud dans une ligne en C++



Nous donnons le coefficient $\beta_{neighbor}$ à la prise de décisions des voisins et le coefficient β_{idio} au choc idiosyncratique. Par conséquent, au cours de chaque moment t , avant le commerce des actions et au cours de chaque tour de consultation, chaque agent reçoit un signal agrégé $Y(t)$:

$$Y_i(\bar{t}) = \beta_{neighbor} * \frac{\sum_{i,j} V_{ij} D(\bar{t})}{N} + \beta_{idio} * C_i(t) \quad (4.1)$$

N est le nombre des voisins connectés avec l'agent i ; i, j indique que la somme est reprise sur l'ensemble des voisins les plus proches de l'agent i ; V_{ij} mesure l'influence qui s'exerce sur l'agent i par la décision $D(\bar{t})$ de son voisin j . Si tous les $V_{ij} = 0$, cela signifie que les agents ne reçoivent aucune information de son voisin, alors qu'ils ne sont pas corrélés du tout. Si tous les $V_{ij} = 1$, tous les agents prennent la même décision, ils achètent tous ou vendent tous une action en même temps. Cela cause alors une grande fluctuation sur le marché financier. En réalité, bien que le marché boursier ait de grandes fluctuations de temps en temps, elles ne suffisent pas pour conduire à un effondrement complet du marché boursier. Cette situation n'est d'ailleurs jamais apparue. En effet, dans les modèles de percolation, les chercheurs ont toujours introduit une probabilité p pour décrire la connexion du voisinage.

La décision de l'agent i dépend de la comparaison entre le signal qu'il reçoit $\sum_{i,j} V_{ij} D(\bar{t})$ et son seuil individuel $\varepsilon_i(t)$. Nous l'avons réalisé par la commande suivante en C++ :

```
const float y = beta_neighbour*neighbour_decision + beta_idio
    *shock[n];
decision[n] = (y < -threshold) ? -1 : (y <= threshold) ? 0 : 1;
```

Cela signifie que si la valeur $Y_i(\bar{t})$ est inférieure à celle du *-threshold*, la décision prise pour l'agent i est -1 (vente d'une action); si la valeur $Y_i(\bar{t})$ est supérieure ou égale à la valeur du *-threshold* et inférieure ou égale à celle du *threshold*, la décision prise par l'agent i est 0 (ni achat ni vente); si la valeur $Y_i(\bar{t})$ est supérieure à la valeur du *threshold*, la décision prise par l'agent i est 1 (achat d'une action). La formule est exprimée mathématiquement en équation comme suit :

$$D(\bar{t}) = -1, \quad \text{si } Y_i(\bar{t}) \leq -\varepsilon_i(t) \quad (4.2)$$

$$D(\bar{t}) = 0, \quad \text{si } -\varepsilon_i(t) < Y_i(\bar{t}) < \varepsilon_i(t) \quad (4.3)$$

$$D(\bar{t}) = 1, \quad \text{si } Y_i(\bar{t}) \geq \varepsilon_i(t) \quad (4.4)$$

IORI (2002) a proposé un modèle similaire. Dans son modèle, la valeur de $\varepsilon_i(t)$ initiale est choisie à partir d'une distribution gaussienne, avec une variance initiale $\sigma_\varepsilon(0)$ et une moyenne nulle. L'hétérogénéité des agents entre à partir de la distribution de $\varepsilon_i(t)$. Pour chaque agent, la valeur de $\varepsilon_i(t)$ est maintenue constante tout au long de la période de commerce, mais elle est ajustée à la fin de cette période avec le mouvement du cours d'une action. Dans notre modèle, nous donnons une valeur constante $\varepsilon_i(t) = 1$ pour éviter de compliquer notre modèle. De plus, ce dernier prend également en compte des circonstances particulières.

Si un agent i a pris une décision d'achat de l'action 1, mais que sur son compte, il n'y a pas assez d'argent, cette décision n'est pas réalisable, alors la valeur de retour de la décision est 0 (ni achat, ni vente). P_{t-1} est le prix de cette action au moment $t-1$.

```
if ((decision[n] == 1) && (money[n] < Ptm1))
    decision[n] == 0;
```

Si un agent i a pris la décision de vendre l'action 1, mais que sur son compte, il n'y a pas assez de stocks, cette décision n'est pas réalisable, alors la valeur de retour de la décision est aussi 0 (ni achat, ni vente).

```
if ((decision[n] == -1) && (stock[n] == 0))
    decision[n] == 0;
```

Si la décision prise n'est pas la même que la décision précédente, alors :

```
if ((decision[n] != previous_decision[n])
    fixed_point_not_obtained = true;
    previous_decision[n] == decision[n] ;
```

Les agents dont le signal idiosyncratique dépasse leurs seuils individuels prennent une décision d'achat ou de vente et cette décision influence ensuite ses voisins selon les équations (4.2), (4.3) et (4.4). La décision de chaque agent est mise à jour successive-

ment suivant la règle, dans les équations (4.1), (4.2), (4.3) et (4.4). En maintenant la valeur de $C_i(t)$ et $\varepsilon_i(t)$, $D(\bar{t})$ et $Y_i(\bar{t})$ interagissent jusqu'à ce qu'ils convergent vers une valeur finale.

4.1.3 Formation du cours boursier

Une fois que le processus de décision a convergé, les agents placent leurs commandes simultanément vers un *market maker*. Le *market maker* détermine la demande agrégative D , l'offre agrégative S d'une action au moment t . Les commandes sont ci-après :

```
const UInt aggregate_demand = count(RANGE(decision), 1);
const UInt aggregate_supply = count(RANGE(decision), -1);
```

Selon la demande agrégative D , l'offre agrégative S , nous pouvons calculer le volume des opérations $V = \frac{D+S}{2}$

```
const double trading_volume = (aggregate_demand +
    aggregate_supply)/2.
```

L'excès de demande relative E est calculé ci-dessous.

```
const double relative_excess_demand=(trading_volume==0.)
    ? 0. : (double(aggregate\_demand) - double(
    aggregate_supply)) / trading_volume ;
```

Cette équation signifie que la valeur de E dépend du volume d'opérations V . Si le $V = 0$, il n'y a pas de volume d'opérations sur le marché, alors la valeur de E est aussi 0; si le V n'est pas égal à 0, la valeur de $E = \frac{D-S}{V}$.

Une vitesse d'ajustement du prix λ est plantée dans le modèle pour calculer le prix actuel de l'action. Si la valeur de E est supérieure à 0,

$$P_t = P_{t-1} * (1 + \lambda * E) \quad (4.5)$$

Si la valeur de E est inférieure à 0,

$$P_t = P_{t-1} * (1 - \lambda * E) \quad (4.6)$$

Si la valeur de E est égale à 0, le prix ne change pas, alors $P_t = P_{t-1}$.

λ est à un intervalle de (0,1). Pour assurer que λ est positive, nous avons introduit λ_i qui suit une distribution de probabilité inverse dans le programme de C++

$$\lambda = \frac{\epsilon(\lambda_i)}{1 + \epsilon(\lambda_i)} \quad (4.7)$$

4.1.4 Transactions pour chaque agent

Une fois que le réseau, le mécanisme de la prise de décision, la demande agrégative, l'offre agrégative et le prix de l'actions sont fixés, nous pouvons créer le processus de transaction pour chaque agent. Il y a trois états pour une transaction, d'après la relation entre la demande agrégative et l'offre agrégative.

Si la demande agrégative est juste égale à l'offre agrégative Lorsque toutes les offres répondent justement aux besoins de toutes les demandes, tous les agents qui ont pris une décision d'achat peuvent acheter une action et tous ceux qui ont pris une décision de vente peuvent vendre une action. Pour un agent i qui a pris une décision d'achat, son stock

va augmenter d'un titre, et son compte va diminuer du montant du prix d'une action. En revanche, pour un agent qui a pris une décision de vente, son stock va diminuer d'un titre et son compte va augmenter du montant du prix d'une action. L'équation suivante nous explique ce processus :

```

if (aggragate_demand == aggregate_supply) {
    for (UInt n = 0; n < N; ++n) {
        stock[n] += decision[n] ;
        money[n] -= decision[n]*Pt ; }
    }

```

Notons ici que le budget est toujours positif. Nous ne considérons pas l'état lorsque le budget est inférieur à zéro. Cela veut dire que l'un des agents ne peut acheter une action à crédit et n'a pas d'autres moyens d'emprunter de l'argent. Quand son compte ne suffit pas pour acheter une action avec le prix Pt , même que les décisions de ses voisins lui donnent un signal d'achat, sa prise de sa décision est un retour à 0, il n'y a alors ni achat ni vente.

Si la demande agrégative est supérieure à l'offre agrégative Tous les agents qui veulent vendre l'action peuvent être satisfaits. Leurs stocks vont diminuer et leur argent augmenter.

```

if (decision[n]) == -1) {
    — stock[n] ;
    money[n] += Pt ;
}

```

Quant aux agents qui veulent acheter l'action, il nous faut introduire une règle de rationnement : d'abord nous devons collecter sur une liste tous les demandeurs, puis réorganiser cette liste ; à la fin nous ordonnons les agents satisfaits de cette liste : du

premier au dernier agent pourtant être satisfait. En réalité, sur un vrai marché financier, cette liste est organisée selon le prix et le volume qu'offre un demandeur. Mais dans notre modèle, nous considérons que tous les demandeurs donnent le même prix et achètent une seule action à la fois. Ainsi, la liste dans notre modèle est organisée par un ordre aléatoire pour tous les demandeurs.

```

else if (decision[n] ==1) ;
rationed_individuals.emplace_back(n) ;
assert (rationed_individuals.size() == aggregate_demand
);

```

Les commandes au-dessus nous montre que si la décision d'un agent n est égale à 1, un signal d'achat, nous ajoutons une valeur n qui correspond à l'agent n derrière la valeur du demandeur temporaire. Nous affirmons que la taille des demandeurs est égale à la valeur de la demande agrégative. Après nous réorganisons la liste pour tous les demandeurs d'après un ordre aléatoire avec la commande *shuffle*. *rng* est un *random number generator*. Cela signifie que l'on utilise un générateur aléatoire pour réaliser un arrangement aléatoire pour tous les agents qui vont acheter une action.

```

shuffle (rationed_individuals.begin() ,
rationed_individuals.end() , rng)

```

Enfin, nous décrivons le processus d'achat pour les demandeurs. Leurs stocks vont augmenter, leurs budgets vont diminuer.

```

for (Unit i=0 ; i < aggregate_supply ; ++i) {
    Unit const n=rationed_individuals[i] ;
    ++ stock[n] ;
    money[n] -=Pt ;
}

```

Si la demande agrégative est inférieure à l'offre agrégative Ce processus est le même que le processus précédent. La seule différence est que tous les agents qui veulent acheter l'action peuvent être satisfaits. Leurs stocks vont augmenter et leurs budgets vont diminuer.

```

if (decision[n] == 1) {
    ++ stock[n] ;
    money[n] -= Pt ;
}

```

Et nous devons aussi utiliser un générateur aléatoire pour faire un arrangement aléatoire pour tous les agents qui vont vendre une action. Nous affirmons que la taille des fournisseurs est égale à la valeur d'offre agrégative.

```

else if (decision[n] == -1) ;
rationed_individuals.emplace_back(n) ;
assert (rationed_individuals.size() == aggregate_supply
)\ ;
shuffle (rationed_individuals.begin(),
rationed_individuals.end(), rng)

```

Pour les fournisseurs, leurs stocks vont diminuer, leurs budgets vont augmenter.

```

for (Unit i=0 ; i < aggregate_demand ; ++i) {
Unit const n=rationed_individuals[i] ;
— stock[n] ;
money[n] +=Pt ;
}

```

4.1.5 Premiers résultats et problèmes rencontrés

Une fois la structure de ce mécanisme de transaction construite, nous pouvons faire la simulation avec une méthode d'optimisation non-linéaire. Mais nous avons rencontré nombre de problèmes.

D'abord, il a fallu trouver un réseau fixe. Au début, nous avons introduit un paramètre p qui décrit une probabilité de connexion entre les deux agents à côtés. Mais, le réseau qui comprend tous les agents et qui décrit leurs positions change avec les différentes valeurs de p . Cela signifie qu'à chaque fois, on obtient un résultat différent à cause du problème de changement de réseau. Si p est supérieur à 75%, tout le réseau a un seul clustering, c'est-à-dire que tout le monde est connecté. Si p est inférieur à 15%, le réseau possède trop de clustering, c'est-à-dire que presque tous les agents sont indépendants, il n'y a pas de propagation d'information entre eux. Mais comment choisir la valeur de p ?

Deuxièmement, nous n'avons pas pu reproduire la queue épaisse et la volatilité en même temps. Dans notre modèle, l'interaction entre les agents est une étape très importante. Nous avons utilisé les trois sources d'information disponibles pour qu'un agent forme son jugement : une source macro-économique, paramètre β_m ; une source liée à son voisinage, paramètre β_n ; et une source idiosyncratique, β_i . Nous avons essayé différentes combinaisons de ces trois sources. Par exemple :

$$feeling = \beta_m * P_{t-1} + \beta_n * feeling[i] + \beta_i * \epsilon_i \quad (4.8)$$

$$feeling = \beta_m * |P_{t-1}| + \beta_n * feeling[i] + \beta_i * \epsilon_i \quad (4.9)$$

$$feeling = \beta_m * P_{t-1}^2 + \beta_n * feeling[i] + \beta_i * \epsilon_i \quad (4.10)$$

$$feeling = \beta_m * P_{t-2} + \beta_n * feeling[i] + \beta_i * \epsilon_i \quad (4.11)$$

$$feeling = \beta_m * P_{t-2}^2 + \beta_n * feeling[i] + \beta_i * \epsilon_i \quad (4.12)$$

$$feeling = \beta_m * |P_{t-1} - P_{t-2}| + \beta_n * feeling[i] + \beta_i * \epsilon_i \quad (4.13)$$

$$feeling = \beta_m * (P_{t-1} - P_{t-2})^2 + \beta_n * feeling[i] + \beta_i * \epsilon_i \quad (4.14)$$

$$feeling = \beta_m * (P_{t-1} - P_{t-2})^3 + \beta_n * feeling[i] + \beta_i * \epsilon_i \quad (4.15)$$

Vu les premiers résultats, avec la source liée à son voisinage β_n et la source idiosyncratique β_i , nous avons pu fabriquer la queue épaisse, avec la valeur de K égale à celle du CAC 40. Mais quand nous avons voulu provoquer la volatilité en introduisant une source macro-économique β_m qui contrôle le changement du prix d'une action, un problème est survenu. Une fois que la valeur de la volatilité a convergé vers la valeur du CAC 40, le rendement moyen étant toujours positif, contrairement à la théorie classique. Nous espérons que le rendement moyen du prix serait nul et que le rendement du lag 1 le serait aussi. Si le rendement moyen n'est pas égal à 0, cela indique que le marché financier n'est plus un jeu à somme nulle. Ensuite, il nous a fallu tenir compte du taux d'inflation, des taux d'imposition (des plus-values), des taux des commissions, etc. Cela a compliqué notre modèle. Nous avons souhaité construire un modèle assez simple, telle est notre but.

Troisièmement, le contrôle de volume des transactions. Parfois nous avons pu reproduire la queue épaisse et la volatilité au hasard. Mais, quand nous avons vérifié les détails de chaque étape, nous avons trouvé que, soit le pourcentage du volume des transactions était proche de 0, soit qu'il était proche de 1, ce qui ne serait pas un bon résultat. Nous

avons vérifié et trouvé que le *trading volume average* de 2013 à 2017 de *S&P 500* était toujours entre 0,6% et 0,8%. Vu que la taille du *CAC 40* est inférieure à *S&P500*, et que son activité est moins forte, nous avons pensé que le *trading volume average* ne dépassait pas 0,6%. Nous avons aussi choisi une entreprise française, Le Group Vinci, pour vérifier cela. En 2016, le *average daily trading volume* était 1 650 000, à la fin de cette année, le nombre total de *share* est 589 000 000. Nous ne savons pas si ce chiffre était le flottant ou si c'était le total. Dans les deux cas, le *trading volume* calculé était 0,27% au maximum. Dans notre modèle, pour assurer que tout était symétrique, nous voulions que la demande soit toujours égale à l'offre, donc un marché en équilibre. (Cela est contraire au marché réel, dont la demande est toujours inférieure à l'offre, 40% contre 60%, jamais en équilibre.)

Pour résoudre les problèmes mentionnés ci-dessous, et pour répondre à la question suivante, *le modèle basé-agents peut-il reproduire les anomalies du marché financier ?*, nous avons apporté une grande modification à notre modèle initial. De plus, nous avons effectué notre simulation avec des méthodes d'optimisation différentes non-linéaires.

4.2 Amélioration du modèle et des résultats : le mimétisme des anticipations permet-il de reproduire les anomalies observées sur les marchés financiers ?

dirigé avec François LEGENDRE

Sur les marchés boursiers, les cours des actions seraient distribués comme une marche au hasard (WALTER, 2006), ce qui n'infirmes pas l'hypothèse d'anticipations rationnelles, même si la volatilité des cours peut sembler « excessive » au regard de la volatilité de leurs valeurs fondamentales. Les cours boursiers présentent toutefois deux grandes anomalies,

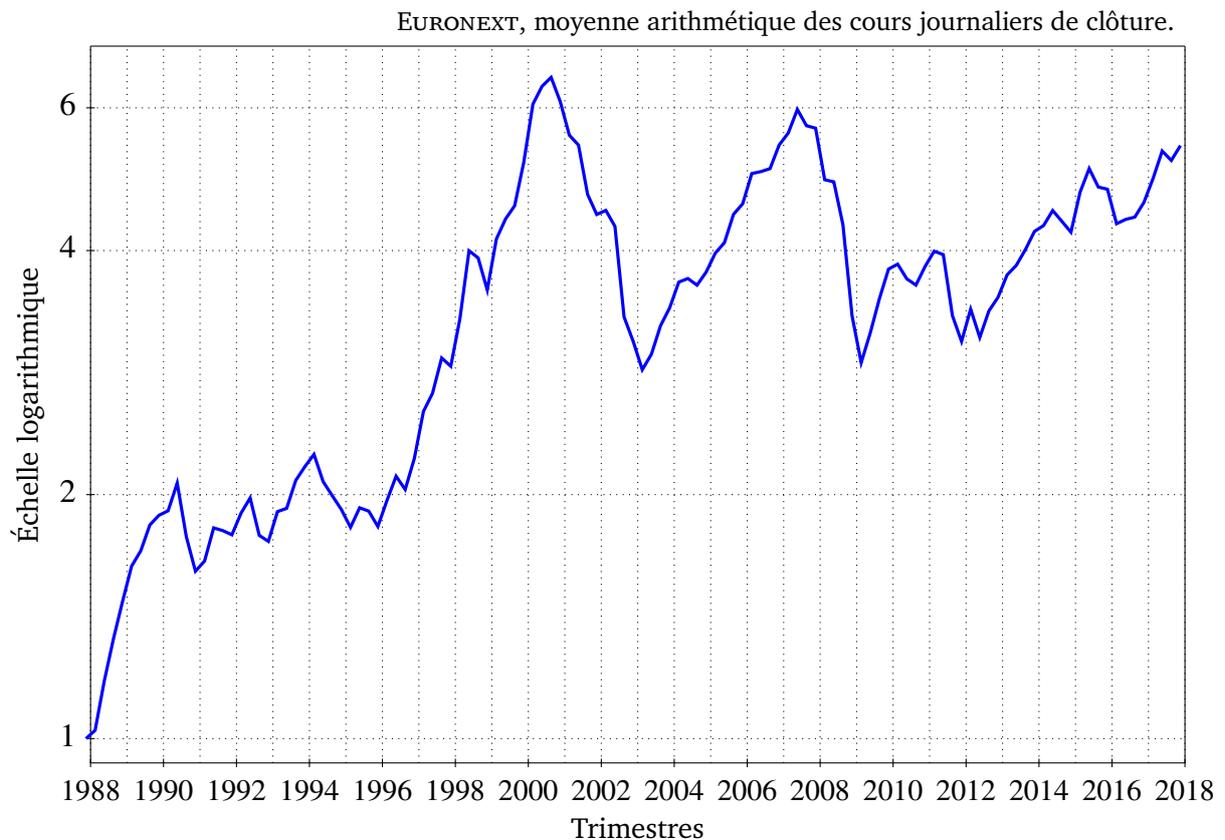
par rapport à une norme qui voudrait que les rendements¹ soient distribués comme une loi normale stationnaire en espérance et en variance. En premier lieu, les rendements présentent une distribution leptokurtique, c'est-à-dire que la partie centrale de la distribution serait « trop » pointue et que, en même temps, les queues de la distribution seraient « trop » épaisses. En second lieu, la volatilité des cours est irrégulière : des périodes de fortes turbulences alterneraient avec des périodes de calmes plats ; cette caractéristique est appelée en anglais *volatility clustering* que nous traduirons par volatilité par grappe. Ces deux anomalies sont documentées depuis longtemps². Les travaux de MANDELBROT (par exemple, MANDELBROT (1963)), s'opposant notamment à MARKOWITZ qui jette à cette époque les bases de la finance moderne standard, mettent en évidence que la distribution des rendements des valeurs spéculatives n'a pas la forme d'une distribution normale et que la probabilité de survenue de valeurs extrêmes est plus élevée que celle prévue par la loi normale. MANDELBROT insiste aussi sur le fait que les aléas les plus importants sont concentrés dans le temps. Cette seconde anomalie peut aussi être mise en évidence en faisant valoir une forme d'hétéroscédasticité des rendements au cours du temps. DING, GRANGER et ENGLE (1993) évoquent ainsi une propriété de « mémoire longue » en observant que, si les rendements ne présentent pas de corrélation temporelle, les carrés des rendements font montre d'une forte persistante au cours du temps.

En niveau, les cours présentent ainsi des évolutions qui seraient compatibles avec l'existence de bulles spéculatives. Sur la figure 4.5, nous avons porté les évolutions du niveau de l'indice boursier CAC 40. On voit que ces évolutions peuvent se décrire en termes de phases d'envolées soutenues qui se ponctuent par un retournement brutal : des bulles se formeraient sur les marchés financiers, bulles appelées nécessairement à éclater. Les marchés financiers seraient ainsi caractérisés par une forte instabilité intrinsèque.

1. Si le niveau du cours d'une action, au temps t , est noté P_t , le rendement de l'action est égal à $\Delta P_t/P_{t-1}$, la plus ou moins value rapportée au montant investi. Ce rendement est souvent approximé par la différence logarithmique : $\Delta P_t/P_{t-1} \approx \ln(P_t) - \ln(P_{t-1})$. L'hypothèse de marche au hasard, en logarithme, s'exprime par la relation $\ln(P_t) = \ln(P_{t-1}) + \varepsilon_t$ où ε_t est un « bruit blanc », c'est-à-dire un processus aléatoire indépendamment et identiquement distribué.

2. Nous ne prétendons pas détailler toutes les singularités statistiques que les cours boursiers présentent ; CONT (2001) et LEBARON (2006) constituent deux bonnes revues sur ce thème.

FIGURE 4.5 : Évolution trimestrielle de l'indice boursier français CAC 40 depuis sa création



Par ailleurs et plus particulièrement depuis la crise financière de 2007, il est relevé (voir, par exemple, COLANDER, HOWITT, KIRMAN, LEIJONHUFVUD et MEHRLING (2008), voir aussi BOOKSTABER (2012) sur l'instabilité financière) que les développements théoriques en macroéconomie qui recherchent des fondements microéconomiques à la macroéconomie se sont beaucoup plus attachés à étudier les comportements individuels que la structure de leurs interactions. Par exemple, les modèles d'équilibre général dynamiques et stochastiques ont permis de mieux comprendre les motivations des réallocations au cours du temps des ressources d'un « agent représentatif » ; par contre, sans doute parce qu'ils restent prisonniers d'une forme très forte de l'hypothèse d'anticipations rationnelles, ils ne permettent guère de décrire les interactions entre individus qui seraient hétérogènes. COLANDER, HOWITT, KIRMAN, LEIJONHUFVUD et MEHRLING

(2008) plaident en conséquence pour le développement de *modèles basés agents*³ de sorte, notamment, à étudier les conséquences d'hypothèses plus réalistes qui porteraient sur l'hétérogénéité des agents, sur les structures d'interactions entre les agents et, enfin, sur le mode de formation des anticipations en particulier par apprentissage. Cet appel n'a toutefois pas vraiment été entendu et les *modèles basés agents* n'ont pas connu, ces dix dernières années, de développements remarquables. Ces modèles peinent à s'imposer, nous semble-t-il, parce qu'ils ne sont pas assez simples et parce qu'ils sont insuffisamment validés. Nous reviendrons sur ce point plus loin mais nos motivations sont ainsi très précises. Nous développons un *modèle basé agents* afin de tester l'hypothèse de mimétisme sur les marchés financiers ; ce modèle est simple ; il est estimé par une méthode de moments simulés afin d'en obtenir une certaine force probante.

Le mimétisme des opinions ne désigne pas nécessairement un comportement mou-tonnier. Il est facile de donner des fondements rationnels au mimétisme. Dans la veine du concours de beauté de KEYNES, il est déjà possible de faire valoir qu'il est préférable, notamment pour un gestionnaire de fonds, de se tromper avec tout le monde que d'avoir tort contre tous. Il est aussi possible d'invoquer le coût élevé d'accès à l'information : dans ce cas, il n'est pas irrationnel de s'appuyer sur les opinions des personnes avec lesquelles il est possible de communiquer. Le mimétisme reste toutefois difficile à établir : on observe effectivement une forte corrélation des comportements d'intervenants qui appartiennent à un même groupe mais cette corrélation pourrait aussi s'expliquer soit par le fait que les membres du groupe accèdent à la même information privée soit par le fait que le groupe est constitué d'individus dont les comportements sont homogènes.

Notre travail est organisé en trois parties. Nous présentons, dans un premier temps, le modèle dit de percolation, notamment popularisé par CONT et BOUCHAUD (2000), ainsi que deux développements marquants dans le domaine des *modèles basés agents* pour étudier les comportements mimétiques en finance. Nous proposons une critique de ses

3. Ces modèles sont très clairement présentés dans TESFATSION (2006) ; voir aussi DE GRAUWE (2010) ou BALLOT, MANDEL et VIGNES (2015).

développements qui pourrait permettre, d'une part, de comprendre leurs faiblesses et, d'autre part, de proposer une modélisation qui puisse donner une force probante aux *modèles basés agents*. Dans un deuxième temps, nous présentons notre modèle qui s'efforce d'être simple, qui s'inscrit dans le prolongement du modèle de percolation et qui cherche à retracer précisément les trois sources d'information que les agents pourraient mobiliser pour former leur propre opinion : une source idiosyncratique, une source provenant de leur voisinage et, en dernier lieu, une source macroéconomique. Dans un dernier temps, nous discutons de la méthode économétrique mise en œuvre, une méthode de moments simulés, et nous présentons les résultats obtenus.

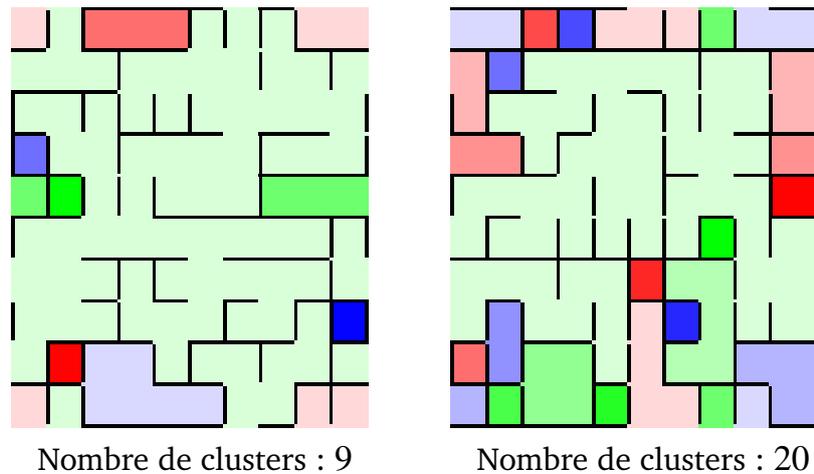
4.3 Le modèle de percolation

Le modèle de percolation a été développé, dans le domaine de la physique des matériaux, pour modéliser la diffusion d'un gaz ou d'un liquide au sein d'un matériau poreux. Il a été repris pour représenter la diffusion d'opinions sur un réseau d'agents et, partant, une formation des anticipations qui relèverait pour partie de l'hypothèse de mimétisme. Nous présentons tout d'abord le travail de CONT et BOUCHAUD (2000), le travail qui a permis de populariser ce modèle pour expliquer la leptokurticité de la distribution des rendements. Nous détaillons dans un second point deux développements récents dans le domaine des *modèles basés agents*.

4.3.1 Le modèle de CONT et BOUCHAUD (2000)

Dans le modèle de CONT et BOUCHAUD (2000), les agents sont placés sur un tore : en dimension 1, ce tore serait un cercle ; en dimension 2, la surface d'un globe. Il est ainsi possible de définir une distance entre deux agents et de construire un réseau entre les agents. Dans le modèle le plus simple, les agents ne peuvent communiquer qu'avec les

FIGURE 4.6 : Deux topologies du réseau de communication entre les agents dans le modèle de percolation de dimension 2 avec $p = 0,45$



agents qui leurs sont adjacents et le réseau est constitué en supposant que la probabilité, notée p , que deux agents adjacents soient connectés est constante. Nous avons porté, sur la figure 4.6, deux topologies du réseau de communication entre les agents, fabriquées au hasard en prenant une probabilité de connexion égale à 0,45 pour un tore de dimension 2. La taille du côté du tore est égale à 10 et le nombre d'agents est ainsi égal à 100. Nous avons matérialisé l'absence de communication entre deux agents par un trait, vertical ou horizontal, entre les deux cellules adjacentes occupées par les deux agents. Les agents situés sur la première ligne (au nord du carré) communiquent avec les agents situés sur la dernière ligne (au sud du carré) ; de même pour les agents situés sur la première et la dernière colonnes : la propriété de tore est ainsi vérifiée.

On voit que le modèle de percolation est un moyen simple de constituer des *clusters*, c'est-à-dire des groupes d'individus connectés, directement ou indirectement. Nous avons figuré ces *clusters* en utilisant des couleurs différentes, d'autant plus foncées que le *cluster* est de petite taille. On observe, en outre, que la distribution de la taille des *clusters* est singulière : quelques *clusters* de grande taille, peu de *clusters* de taille moyenne et, enfin, de plus nombreux *clusters* de petite taille. Cette distribution est très dépendante de la probabilité de connexion : il existe une probabilité critique pour laquelle on est quasiment assuré d'obtenir un *cluster* unique qui rassemble tous les individus du tore.

CONT et BOUCHAUD (2000) retiennent alors les trois hypothèses simplificatrices suivantes. En premier lieu, chaque agent, repéré par un indice inférieur i , ne va pouvoir réaliser au temps t , que l'une de ces trois possibilités : soit passer un ordre de vente pour une unité de l'actif financier ($\phi_i^t = -1$), soit ne rien faire ($\phi_i^t = 0$), soit passer un ordre d'achat pour une unité de l'actif ($\phi_i^t = 1$). Aussi la demande excédentaire D_t au temps t s'exprime-t-elle comme suit :

$$D_t = \sum_{i=1}^N \phi_i^t \quad (4.16)$$

où N est le nombre total d'agents. En deuxième lieu, les agents d'un même *cluster*, parce qu'ils peuvent librement communiquer entre eux, retiennent tous la même possibilité. En repérant par α le *cluster*, on a ainsi :

$$\phi_i^t = \phi_\alpha^t \quad \forall i \in I_\alpha \quad (4.17)$$

où I_α est l'ensemble des indices des agents qui appartiennent au *cluster* α . Avec cette hypothèse, la demande excédentaire se réécrit de la façon suivante :

$$D_t = \sum_{\alpha=1}^k W_\alpha \phi_\alpha^t \quad (4.18)$$

où k est le nombre de *clusters* et W_α le nombre d'agents du *cluster* α . En troisième et dernier lieu, il est supposé la présence d'un teneur de marché qui fait varier les prix en fonction de la demande excédentaire comme suit

$$\Delta P_t = \frac{1}{\lambda} D_t = \frac{1}{\lambda} \sum_{\alpha=1}^k W_\alpha \phi_\alpha^t \quad (4.19)$$

où le paramètre λ s'interprète comme la profondeur du marché.

CONT et BOUCHAUD (2000) montrent alors que la structure de leur modèle s'apparente à un graphe aléatoire, étudié par les mathématiciens au début des années 1960. En économie, ce serait KIRMAN (1983) qui aurait le premier suggéré d'utiliser cette approche

pour étudier, notamment, la transmission de l'information dans le cœur d'une économie dans une perspective d'équilibre général. Les queues de la distribution asymptotique (en supposant que le nombre d'agents tend vers l'infini mais que la probabilité de connexion dans le tore diminue de sorte que chaque agent soit connecté en moyenne à au plus un autre agent) sont alors plus épaisses que les queues d'une distribution normale.

Comme les auteurs le relèvent eux-mêmes, le modèle ne permet pas de valider l'hypothèse de mimétisme : le résultat est bien sûr lié à la topologie du réseau engendré par le modèle de percolation mais la distribution de la taille des clusters pourrait être identifiée, par exemple, à la distribution de la taille des intervenants sur les marchés. Il faudrait aller plus en amont pour proposer une modélisation structurelle de la formation des opinions où les agents échangent de l'information selon la structure proposée par le modèle de percolation.

Ce modèle a engendré une littérature importante, à la fois au plan de la théorie statistique (par exemple, sur la robustesse du résultat vis-à-vis de la dimension du tore) et au plan des vérifications empiriques (en lien notamment avec la microstructure des marchés financiers). Il reste l'un des modèles les plus simples pour engendrer une distribution leptokurtique des rendements. Nous présentons maintenant deux développements marquants de ce modèle pour illustrer les directions prises dans le domaine des *modèles basés agents*.

4.3.2 Deux développements récents des *modèles basés agents* en finance

Nous présentons les développements de IORI (2002) et de FABRETTI (2013) qui illustrent bien, nous semble-t-il, les directions récentes. Le modèle de IORI (2002) se propose, explicitement, d'enrichir le modèle de CONT et BOUCHAUD (2000). Elle développe pour cela un cadre structurel précis pour représenter les échanges d'information entre les

agents : ceux-ci sont placés sur un tore carré de taille $L \times L$ et ils communiquent avec leurs quatre voisins immédiats. Chaque période, notée t , est décomposée en plusieurs tours d'échanges d'information, notés \tilde{t} , si bien que $Y_i(\tilde{t})$, le signal reçu par un agent au cours du tour \tilde{t} , s'exprime en fonction des décisions $S_j(\tilde{t})$ prises (provisoirement) par les autres agents comme suit :

$$Y_i(\tilde{t}) = \sum_{j \in N(i)} J_{ij} S_j(\tilde{t}) + A \nu_i(t) \quad (4.20)$$

où $N(i)$ est l'ensemble des indices du voisinage de l'agent i , J_{ij} une mesure de l'influence des décisions du voisin j sur le signal de l'agent i , A un paramètre d'échelle et $\nu_i(t)$ un signal aléatoire idiosyncratique à i fixé au début de la période t .

La décision de l'agent i résulte du signal et d'un seuil ξ_i propre à l'agent i

$$S_i(\tilde{t}) = \begin{cases} -1 & \text{si } Y_i(\tilde{t}) \leq -\xi_i(t) \\ 0 & \text{si } -\xi_i(t) < Y_i(\tilde{t}) < \xi_i(t) \\ 1 & \text{si } \xi_i(t) \leq Y_i(\tilde{t}) \end{cases} \quad (4.21)$$

où, comme dans CONT et BOUCHAUD (2000), la décision prend trois valeurs : -1 (ordre de vente), 0 (inaction) et 1 (ordre d'achat). Le seuil $\xi_i(t)$ est fondé par un coût de transaction ; ce seuil est fixé au début pour chaque agent en étant tiré dans une loi normale pour retracer une forme d'hétérogénéité persistante entre les agents. Le seuil est révisé à chaque période de sorte que sa valeur, proportionnellement au prix de l'actif, soit constante :

$$\xi_i(t) = \frac{P(t)}{P(t-1)} \xi_i(t-1) \quad \forall i \quad (4.22)$$

La spécification retenue est ainsi particulière puisque les coûts de transaction, quand il s'agit de retracer des frictions aux échanges, sont habituellement considérés comme fixes et non spécifiques.

Le prix ne varie pas en fonction de la demande excédentaire, mais de la tension sur le marché définie comme le rapport du niveau de la demande agrégée $D(t)$ et du niveau de l'offre agrégée $Z(t)$:

$$D(t) = \sum_{i / S_i(t)=1} S_i(t) \quad Z(t) = - \sum_{i / S_i(t)=-1} S_i(t) \quad \frac{P(t)}{P(t-1)} = \left(\frac{D(t)}{Z(t)} \right)^\alpha \quad \text{avec } \alpha > 0 \quad (4.23)$$

où le paramètre α s'interprète comme la vitesse d'ajustement du prix. Ce paramètre dépend lui-même de l'intensité relative des transactions :

$$\alpha = a \frac{D(t) + Z(t)}{L^2} \quad \text{avec } a > 0 \quad \text{et} \quad 0 \leq \frac{D(t) + Z(t)}{L^2} \leq 1 \quad (4.24)$$

Enfin, chaque agent est initialement doté d'une encaisse monétaire et d'un stock de titres financiers. Ces deux stocks sont mis à jour au cours de la simulation et leurs niveaux effectifs constituent deux contraintes supplémentaires qui viennent réduire, pour chaque agent, les possibilités de vente et d'achat du titre financier. Le modèle est ainsi relativement détaillé, sans doute dans un souci de réalisme des hypothèses.

Toutefois, la représentation qui est proposée de la diffusion de l'information entre les agents est un peu singulière. Le signal de l'agent i au tour \tilde{t} , $Y_i(\tilde{t})$, dépend non pas des signaux des voisins [les $Y_j(\tilde{t})$, $j \in N(i)$] mais de leurs décisions virtuelles [les $S_j(\tilde{t})$]. IORI (2002) insiste sur le fait que les tours infra-périodiques permettent de mettre à jour l'information de chaque agent et que ces tours sont itérés tant qu'un point fixe n'est pas obtenu : le nombre de tours est ainsi variable, d'une période à l'autre. Cette représentation s'apparente finalement au tâtonnement walrasien qui s'effectue « à horloge arrêtée »⁴ et qui exige que les agents forment, provisoirement, des offres et des demandes notionnelles.

IORI (2002) valide son modèle en utilisant l'heuristique suivante. Elle suppose, tout d'abord, que les agents ne sont pas connectés ; en simulant le modèle en faisant l'hy-

4. Selon l'expression de D'AUTUME (1982).

pothèse d'une vitesse d'ajustement du prix fixe (le paramètre α est constant et l'équation (4.24) n'est pas utilisée), la propriété de volatilité par grappe est obtenue. Toutefois, une corrélation négative est observée entre cette volatilité et le volume des transactions, ce qui n'est pas conforme aux observations. Dans un deuxième temps, en permettant aux agents de communiquer entre eux, un système très instable est obtenu ; sa stabilisation, si elle provient d'une faible vitesse d'ajustement du prix, ne permet plus d'obtenir la propriété de volatilité par grappe. En faisant dépendre cette vitesse de l'intensité des transactions (l'équation (4.24) est cette fois-ci utilisée), la propriété de volatilité par grappe est retrouvée. Enfin, quand les seuils qui fixent la zone d'inaction restent constants au cours du temps, la propriété de volatilité par grappe disparaît. Au total, le mimétisme permet d'engendrer des déséquilibres importants entre l'offre et la demande et les deux mécanismes de propagation temporelle (la variation des seuils d'inaction et la vitesse d'ajustement paramétrée par l'intensité des transactions) conduisent à l'obtention de la volatilité par grappe (présence d'autocorrélation des carrés ou des valeurs absolues des rendements) tout en conservant la propriété de marche aléatoire (absence d'autocorrélation des rendements). Le modèle est validé, essentiellement, en montrant que l'autocorrélation des valeurs absolues des rendements, obtenu à partir d'une simulation, présente le profil caractéristique d'un processus « à mémoire longue ».

Le modèle parvient ainsi à rendre compte des deux « faits stylisés » qui nous intéressent (la leptokurticité des rendements et leur volatilité par grappe) ; il peine, nous semble-t-il, à être complètement convaincant. La diffusion de l'information entre les agents est instantanée (puisque les tours d'échange entre ceux-ci s'effectuent « à horloge arrêtée ») ; on ne sait pas si les deux mécanismes qui engendrent la volatilité par grappe sont complémentaires ou substituts ; l'heuristique qui cherche à identifier les mécanismes à l'œuvre n'est pas systématique ; enfin, le modèle n'est validé que sommairement. Le second travail dont nous voulons rendre compte s'attache, justement, à mieux valider le modèle en s'inscrivant dans une démarche d'inférence statistique.

FABRETTI (2013) insiste sur la nécessité d'étalonner les *modèles basés agents* et se propose, dans son travail, d'estimer le modèle de FARMER et JOSHI (2002) en spécifiant un critère ayant des fondements statistiques et en utilisant un algorithme de résolution numérique adapté⁵. Le modèle de FARMER et JOSHI (2002) cherche à expliquer la dynamique des prix sur les marchés financiers en distinguant deux sortes d'intervenants : les « fondamentalistes » et les « chartistes ». La motivation reste celle de reproduire les deux anomalies que nous avons relevées.

FABRETTI (2013) plaide pour l'utilisation d'une méthode de moments simulés pour estimer les paramètres du modèle. Soient $\underline{\theta}$ le vecteur (de taille k) des paramètres et $\underline{\hat{m}}$ le vecteur (de taille p) d'une estimation d'un certain nombre de moments qui caractérisent l'évolution du prix d'un actif, comme par exemple l'écart-type estimé du cours (noté $\hat{\sigma}$), son kurtosis estimé (noté $\hat{\kappa}$), etc. Par simulation, en fixant une valeur particulière à $\underline{\theta}$, il est possible d'obtenir une estimation des moments, que l'on note $\underline{\tilde{m}}(\underline{\theta})$. L'estimateur des moments simulés est alors défini comme suit :

$$\hat{\underline{\theta}} = \arg \min_{\underline{\theta}} [\underline{\tilde{m}}(\underline{\theta}) - \underline{\hat{m}}]' W [\underline{\tilde{m}}(\underline{\theta}) - \underline{\hat{m}}] \quad (4.25)$$

où W est une matrice (de taille $p \times p$) symétrique semi-définie positive qui donne la métrique utilisée.

FABRETTI (2013) suggère d'utiliser, comme métrique, la métrique statistiquement optimale (c'est-à-dire la métrique qui fournit asymptotiquement les estimateurs de plus faible variance) en prenant

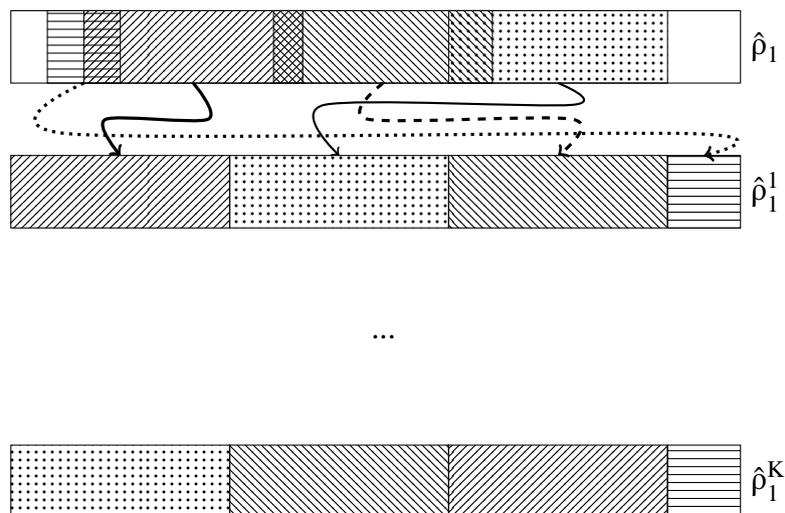
$$W = \hat{\Omega}^{-1} \quad (4.26)$$

où $\hat{\Omega}$ est une estimation de la matrice de variance-covariance des estimateurs des moments :

$$\Omega = V(\underline{\hat{m}}) \quad (4.27)$$

5. FABRETTI (2013) s'appuie sur des travaux antérieurs pour étayer sa proposition dont notamment le recours au *moving block bootstrap*.

FIGURE 4.7 : La méthode du *moving block bootstrap* pour estimer la distribution empirique de $\hat{\rho}_1$, l'estimateur de l'autocorrélation d'ordre 1 d'un processus



Enfin, FABRETTI (2013) propose de calculer $\hat{\Omega}^{-1}$ en recourant à une méthode de *moving block bootstrap*. En effet, certains moments ont pour objet de caractériser des formes de dépendance temporelle et une méthode de simple *bootstrap* ne serait pas en mesure de chiffrer la variance des estimateurs de ces moments. Sur la figure 4.7, nous représentons la méthode qui consiste à tirer, avec remise, non pas une observation comme dans le simple *bootstrap*, mais un bloc d'observations consécutives pour former une nouvelle série. Il est ainsi possible d'obtenir la distribution empirique d'un estimateur comme par exemple l'estimateur de l'autocorrélation d'ordre 1 d'un processus stochastique stationnaire.

On voit, sur ce graphique, que la longueur du bloc qui a été retenue conduit à tirer au hasard quatre blocs pour reconstituer une réalisation de l'échantillon. Le choix de la longueur du bloc résulte d'un arbitrage. D'un côté, il faut prendre un bloc de taille suffisante pour retracer la dépendance temporelle des observations : chaque changement de bloc détruit cette dépendance. De l'autre côté, il ne faut pas prendre un bloc de trop grande taille pour ne pas sous-estimer l'ampleur des fluctuations d'échantillonnage. BÜHLMANN et KÜNSCH (1999) discutent en détails de cet arbitrage ; voir aussi COGNEAU et ZAKAMOULINE (2010).

Dans le travail de FABRETTI (2013), cinq moments seulement sont retenus ; la matrice $\hat{\Omega}$ est obtenue par *moving block bootstrap* en prenant une taille de bloc égale à 100 et 10 000 réplifications à partir de données journalières de l'indice boursier Standard & Poor 500 recueillies pendant deux ans. Le modèle, de son côté, comporte 12 paramètres. Avec les notations que nous avons introduites, on a $k = 12$ et $p = 5$. Le modèle est donc très fortement sous-identifié mais ce point, qui nous semble central, n'est évoqué que brièvement par l'auteure. Par contre, celle-ci discute longuement des méthodes numériques de minimisation du critère des moments généralisés donné par l'équation (4.25) en plaidant, par exemple, pour un algorithme de simplexe proposé par NELDER et MEAD. Les résultats obtenus sont cependant peu probants en étant notamment particulièrement instables.

Nous proposons ainsi d'aller plus en avant dans ces deux voies : d'une part, utiliser le modèle de percolation pour modéliser la diffusion des opinions entre les agents ; d'autre part, estimer les paramètres du modèle par une méthode de moments simulés.

Nous proposons ainsi d'aller plus en avant dans ces deux voies : d'une part, utiliser le modèle de percolation pour modéliser la diffusion des opinions entre les agents ; d'autre part, estimer les paramètres du modèle par une méthode de moments simulés.

4.4 Un modèle simple

Pour tester l'hypothèse de mimétisme, il nous faut modéliser la diffusion des opinions entre les agents ; il nous faut la modéliser de la manière la plus simple possible pour deux raisons différentes. En premier lieu, nous voulons pouvoir attribuer au mimétisme les anomalies que l'on relève sur les marchés financiers. Dans un modèle qui serait trop compliqué, il est beaucoup plus difficile d'imputer à une hypothèse (ou à un jeu d'hypothèse) des conséquences. C'est la critique que nous venons d'adresser au modèle de IORI

(2002) où deux mécanismes permettent d'expliquer la volatilité par grappe sans que l'on puisse vraiment savoir si ces mécanismes sont complémentaires ou substituts. Les plateformes qui permettent de développer des *modèles basés agents* conduisent bien souvent les chercheurs à proposer des modèles « trop » réalistes qui, au final, fonctionnent comme des boîtes noires. Nous cherchons à reproduire, de manière robuste et simple, l'instabilité observée sur les marchés financiers.

En second lieu, le modèle doit être simple pour pouvoir être simulé, sur l'ordinateur, avec une grande efficacité. En effet, il nous faut simuler le modèle un très grand nombre de fois pour réduire la place prise par l'aléa inhérent à la simulation. Ce point est trop peu évoqué, par exemple, dans le travail de FABRETTI (2013) ; nous reviendrons sur cette difficulté plus loin.

Nous reprenons donc le cadre du modèle de percolation. Nous nous étions tenus, dans un premier temps, à une version forte de ce modèle où tous les agents d'un même *cluster*, à l'issue d'un grand nombre d'échanges d'information, se coordonnent sur la même opinion. Nous avons ensuite délaissé cette modélisation pour les trois raisons suivantes. En premier lieu, nous nous sommes rendus compte que cette modélisation n'est pas réaliste : elle suppose que les agents, dans le laps de temps infra-périodique, disposent du temps nécessaire pour uniformiser leur opinion quelle que soit la taille du *cluster* dont ils sont membres. Ensuite, nous subissions la vieille critique faite au modèle de percolation : les effets résultent-ils du mimétisme des opinions qui conduit chaque agent d'un même *cluster* à adopter la même conduite ou résultent-ils de la distribution de la taille des intervenants, chaque cluster représentant finalement un intervenant ? Enfin, il s'avère difficile d'estimer un modèle de percolation. Il faudrait en effet estimer le paramètre p (la probabilité que deux agents adjacents soient connectés) et ce paramètre ne détermine pas la topologie effective du réseau. On le voit par exemple sur la figure 4.6 où la même valeur du paramètre p (en l'occurrence ce paramètre est égal à 0,45) donne naissance à deux configurations très différentes du réseau.

Pour ne donc pas subir la critique selon laquelle les agents se coordonnent « à horloge arrêtée », nous avons supposé un nombre de tours d'échange de l'information fixé, de manière arbitraire, à 5. Pendant une période, les agents ont la possibilité, à cinq reprises, d'échanger avec leurs voisins. Ceux-ci, en effet, sont placés sur un tore de dimension 2 et ils ont ainsi la possibilité de former leurs opinions en prenant connaissance des opinions de leur voisinage. Dans un premier temps, nous avons envisagé différentes formes pour cet échange comme, par exemple, la moyenne des opinions des quatre voisins ou, encore, l'opinion majoritaire. Nous avons finalement décidé de faire simple : chaque agent, à chaque tour, accède à l'opinion de l'un de ses quatre voisins, tiré au hasard.

Pour ne donc pas subir la critique selon laquelle les agents se coordonnent « à horloge arrêtée », nous avons supposé un nombre de tours d'échange de l'information fixé, de manière arbitraire, à 5. Pendant une période, les agents ont la possibilité, à cinq reprises, d'échanger avec leurs voisins. Ceux-ci, en effet, sont placés sur un tore de dimension 2 et ils ont ainsi la possibilité de former leurs opinions en prenant connaissance des opinions de leur voisinage. Dans un premier temps, nous avons envisagé différentes formes pour cet échange comme, par exemple, la moyenne des opinions des quatre voisins ou, encore, l'opinion majoritaire. Nous avons finalement décidé de faire simple : chaque agent, à chaque tour, accède à l'opinion de l'un de ses quatre voisins, tiré au hasard.

Notons ϕ_i^{τ} l'opinion de l'agent i au tour τ de la période t . Nous distinguons trois sources différentes dans la formation de l'opinion des agents. En premier lieu, une source idiosyncratique ; cette source sera simplement spécifiée sous la forme d'un choc propre à chaque agent, tiré dans une loi normale centrée réduite au début de chaque période. Ce choc est noté ε_i^t . En deuxième lieu, comme nous venons de nous en expliquer, nous supposons que chaque agent utilise son voisinage pour former son opinion. Nous désignons par j_i^{τ} l'indice du voisin de l'agent i ; au tour τ de la période t , l'agent i prend connaissance de l'opinion de l'agent j_i^{τ} . Cet agent est tiré au sort, au sein des quatre voisins de l'agent i : celui situé au nord sur le tore, celui situé à l'est, celui situé au sud et, enfin,

celui situé à l'ouest. La dernière source d'information est la source macroéconomique : les agents exploitent cette source mais uniquement avec un décalage d'une période.

Ces trois sources d'information sont paramétrées par les coefficients, respectivement, β_i (pour idiosyncratique), β_v (pour voisinage) et β_m (pour macroéconomique). Au tour 0 de la période t , l'agent i forme une opinion initiale comme suit

$$\phi_i^{t_0} = [1 + |\Delta P_{t-1}|]^{\beta_m} \beta_i \varepsilon_i^t \quad (4.28)$$

où P_t est le cours de l'actif au temps t . Nous justifierons plus bas la forme fonctionnelle pour la composante macroéconomique. Au tour τ de la période t , avec $\tau > 0$, l'agent i met à jour son opinion de la façon suivante

$$\phi_i^{t_\tau} = [1 + |\Delta P_{t-1}|]^{\beta_m} \left[\beta_v \phi_{j_i}^{t_{\tau'}} + \beta_i \varepsilon_i^t \right] \quad \tau = 1, \dots, 5 \quad (4.29)$$

où $\phi_{j_i}^{t_{\tau'}}$ est, soit la nouvelle opinion de l'agent j du tour τ si la mise à jour de l'opinion de l'agent j intervient avant celle de l'agent i , soit l'ancienne opinion de l'agent j du tour $\tau-1$ sinon. L'ordre dans lequel ces mises à jour interviennent est fixé ; il nous est ainsi possible de simuler le modèle très simplement en parcourant la population séquentiellement toujours dans le même ordre et en itérant ce parcours à cinq reprises. Le processus de mise à jour des opinions est en conséquence un protocole séquentiel pour lequel il suffit de disposer de l'opinion courante de chaque agent⁶.

Dans notre modèle, le mimétisme est principalement destiné à engendrer la leptokurticité de la distribution des rendements et la source macro-économique à engendrer la seconde anomalie, la volatilité par grappe. Une spécification simple de la composante

6. Nous ne voulons nous fermer la possibilité de simuler le modèle sur un processeur graphique ; il faut donc pouvoir représenter les agents de la façon la plus compacte possible.

macroéconomique aurait été d'augmenter l'équation d'un terme additif comme suit :

$$\phi_i^{t\tau} = \beta_m(\Delta P_{t-1})^2 + \beta_v \phi_{j_i}^{t\tau'} + \beta_i \varepsilon_i^t \quad \tau = 1, \dots, 5 \quad (4.30)$$

Ainsi, une plus forte volatilité du cours entre $t-1$ et $t-2$ induirait une plus grande volatilité des opinions en t . Cette forme a toutefois l'inconvénient de ne pas conduire à une distribution symétrique des opinions. Nous avons finalement opté pour la spécification de l'équation (4.29) qui inclut un terme multiplicatif fondé sur la valeur absolue de la variation du cours entre $t-1$ et $t-2$. On obtient de la sorte un mécanisme de formation des anticipations qui engendre une distribution symétrique des opinions : la distribution du terme ε_i est symétrique et celle du terme $\phi_{j_i}^{t\tau'}$ pourrait être de la même manière symétrique si on retient l'équation (4.29). Nous proposons ainsi un mécanisme qui pourrait conduire à la persistance de la volatilité du cours boursier. Bien sûr, il faudrait dans un travail ultérieur donner des fondements plus précis à ce canal ; dans un premier temps, nous cherchons, empiriquement, une spécification simple qui puisse être robuste.

À l'issue de ces cinq tours d'échanges des opinions qui figurent ainsi leur diffusion progressive au sein du tore, les agents prennent leur décision. Comme dans CONT et BOUCHAUD (2000), trois décisions alternatives peuvent être prises :

$$d_i^t = \begin{cases} -1 & \text{si } \phi_i^{t5} \leq -1 & \text{ordre de vente d'une unité de l'actif} \\ 0 & \text{si } -1 < \phi_i^{t5} < 1 & \text{inaction} \\ 1 & \text{si } 1 \leq \phi_i^{t5} & \text{ordre d'achat d'une unité de l'actif} \end{cases} \quad (4.31)$$

où d_i^t est la décision de l'agent i en fin de période t et ϕ_i^{t5} l'opinion de l'agent i à l'issue de cinq tours d'échanges d'information. Le choc idiosyncratique, ε_i^t , est de variance unitaire ; le paramètre β_i contribue à fixer la dispersion des opinions ; aussi n'est-il pas nécessaire, à la différence de IORI (2002), de paramétrer la valeur des seuils : on peut se contenter de prendre -1 et 1 .

La demande excédentaire résulte de l'agrégation des offres et des demandes individuelles : La demande excédentaire résulte de l'agrégation des offres et des demandes individuelles :

$$D_t = \sum_{i=1}^N d_i^t \quad (4.32)$$

et le teneur de marché ajuste le prix en fonction de cette demande excédentaire :

$$P_t = P_{t-1} + \frac{1}{\lambda} D_t \quad (4.33)$$

C'est donc ce modèle très simple que nous cherchons à estimer ; ce modèle ne comprend que quatre paramètres qui peuvent chacun faire l'objet d'une interprétation simple :

1. le paramètre β_i donne la contribution, dans la formation des opinions, de la composante idiosyncratique ;
2. le paramètre β_v donne la contribution, dans la formation des opinions, de la composante mimétique ;
3. le paramètre β_m donne la contribution, dans la formation des opinions, de la composante macroéconomique ;
4. le paramètre λ , qui s'interprète comme la profondeur du marché, est essentiellement un paramètre d'échelle qui relie la variation absolue du cours et le niveau de la demande excédentaire.

4.5 Estimation du modèle et résultats

Nous nous proposons d'estimer le modèle que nous venons de présenter par une méthode de moments simulés. Nous discutons, dans un premier temps, des moments qu'il faudrait retenir pour cet exercice avant de présenter, dans un second point, les résultats obtenus.

4.5.1 Quels moments retenir ?

La table 4.1 rassemble un certain nombre de moments que nous avons calculés en utilisant le cours journalier de clôture de l'indice CAC 40. Nous utilisons toutes les données disponibles du 31 décembre 1987 (date de création de l'indice) au 22 décembre 2017 (date à laquelle nous avons fixé les données de cette étude). Les rendements sont calculés en utilisant la différence des logarithmes. Nous laissons de côté deux difficultés. La première a trait à la valeur moyenne des rendements. En effet, nous n'allons pas chercher à reproduire le rendement moyen de l'indice mais seulement des anomalies relatives à la dispersion des rendements. Il ne nous a pas semblé utile d'introduire dans le modèle des facteurs de croissance en supposant par exemple que le nombre d'agents croît régulièrement au cours du temps. D'autre part, nous nous sommes contentés des rendements nominaux et non des rendements réels. Depuis 1988, l'inflation est particulièrement modérée et relativement stable en France ; nous en sommes ainsi restés aux rendements nominaux.

TABLE 4.1 : Quelques moments statistiques caractérisant les rendements de l'indice boursier français journalier CAC 40

Moment	Estimation	Moyenne	Écart-type	Coeff. variation
σ	0,0136	0,0137	0,000 901	0,0660
κ	4,68	4,45	1,18	0,266
ρ_1	-0,002 06	-0,004 22	0,0157	-3,73
ρ_{10}	-0,001 61	0,000 336	0,0161	47,8
χ_1	0,185	0,176	0,0238	0,135
χ_{10}	0,209	0,185	0,0506	0,274

Source : cours journalier de clôture de l'indice CAC 40 du 31 décembre 1987 au 22 décembre 2017, soit 7580 observations et 7579 rendements. Les nombres sont tous présentés intentionnellement avec 3 chiffres.

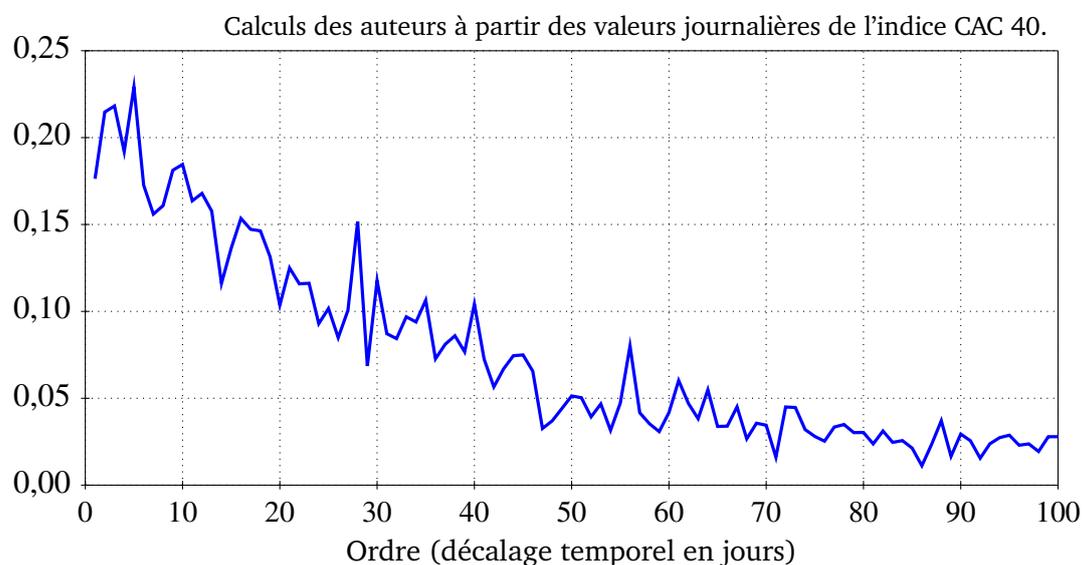
κ est le kurtosis normalisé, c'est-à-dire le moment d'ordre 4 des rendements centrés réduits diminué de 3 ; χ_k est l'autocorrélation d'ordre k (en jours) des carrés des rendements.

La colonne intitulée « Estimation » donne les moments calculés sur l'ensemble de l'échantillon. Les colonnes intitulées « Moyenne », « Écart-type » et « Coeff. variation » donnent les caractéristiques des distributions des estimateurs obtenues par *moving block bootstrap* en prenant une taille de bloc égale à 250 jours et un nombre de répliquions égal à 100 000.

L'indice CAC 40 présente bien les deux anomalies qui nous occupent. Le kurtosis, le moment d'ordre 4 de la variable centrée réduite, est habituellement utilisé pour ca-

racteriser la leptokurticité d'une distribution ; c'est-à-dire à la fois une distribution plus « pointue » en sa moyenne que la loi normale et une distribution dont les queues sont plus épaisses que celles de la loi normale. Dans la table 4.1, nous avons porté le kurtosis normalisé dont la valeur, au cas de la loi normale, est égale à 0. L'estimation obtenue est égale à 4,68 et son écart-type estimé par la méthode du *moving block bootstrap* est égal à 1,18. On voit ainsi que la distribution des rendements est significativement leptokurtique mais que le kurtosis n'est toutefois pas estimé très précisément.

FIGURE 4.8 : Autocorrélations des carrés des rendements de l'indice boursier français journalier CAC 40 en fonction de son ordre



La seconde anomalie, la volatilité par grappe, peut être mesurée par l'autocorrélation des carrés des rendements ; d'aucuns utilisent aussi l'autocorrélation des valeurs absolues des rendements. Nous avons noté χ_1 l'autocorrélation d'ordre 1 des carrés des rendements. Cette autocorrélation prend une valeur élevée, égale à 0,185 ; cette valeur en outre est très significativement différente de 0. La volatilité par grappe n'est toutefois pas seulement caractérisée par l'autocorrélation d'ordre 1 des carrés des rendements ; on parle aussi de « mémoire longue » pour évoquer cette volatilité parce que l'on observe la présence d'autocorrélation des carrés des rendements pour des ordres beaucoup plus élevés. Nous avons représenté, à la figure 4.8, l'autocorrélogramme des carrés des rendements. On voit, effectivement, que les autocorrélations ne décroissent que très len-

tement. Dans la table 4.1, la valeur de l'autocorrélation d'ordre 10 (soit au bout de deux semaines, le marché étant ouvert cinq jours par semaine) est égale à 0,209 ; elle est donc supérieure à l'autocorrélation d'ordre 1 mais elle est estimée moins précisément.

Enfin, les autocorrélations des rendements sont bien toutes nulles : les estimations de ρ_1 et de ρ_{10} (les autocorrélations d'ordre 1 et 10 des rendements) ne sont pas statistiquement différentes de 0. Le CAC 40 suit donc bien une marche au hasard mais dans une version faible puisque la distribution du bruit blanc n'est pas normale et que le processus n'est pas stationnaire au deuxième ordre.

La méthode du *moving block bootstrap* met aussi en évidence que les moments ne sont pas tous estimés avec la même précision. Il apparaît notamment que les moments qui permettent de caractériser les anomalies ne sont pas estimés très précisément. Bien sûr, l'ordre de grandeur des moments peut être très différent et il convient ainsi de ne pas se contenter de l'examen de l'écart-type estimé. C'est pour cette raison que nous avons porté dans la table 4.1 une dernière colonne pour donner le coefficient de variation, c'est-à-dire l'écart-type divisé par la moyenne. Cette quantité est toutefois inappropriée pour ρ_1 et ρ_{10} qui sont logiquement nuls. On voit que le coefficient de variation pour $\hat{\sigma}$ (l'estimateur de l'écart-type des rendements) est quatre fois plus faible que le coefficient pour $\hat{\kappa}$ (l'estimateur du kurtosis des rendements). La précision relative de $\hat{\chi}_{10}$ (l'estimateur de l'autocorrélation d'ordre 10 des carrés des rendements) est comparable à celle de $\hat{\kappa}$.

TABLE 4.2 : Matrice de corrélation des estimateurs des moments des rendements

	$\hat{\sigma}$	$\hat{\kappa}$	$\hat{\rho}_1$	$\hat{\rho}_{10}$	$\hat{\chi}_1$	$\hat{\chi}_{10}$
$\hat{\sigma}$	1					
$\hat{\kappa}$	0,29	1				
$\hat{\rho}_1$	-0,24	-0,58	1			
$\hat{\rho}_{10}$	-0,22	0,10	0,21	1		
$\hat{\chi}_1$	0,39	0,66	-0,39	0,015	1	
$\hat{\chi}_{10}$	0,58	0,68	-0,38	-0,099	0,69	1

Voir les notes du tableau 4.1 ; Les nombres sont tous présentés intentionnellement avec 2 chiffres.

Le dernier intérêt *moving block bootstrap* est de permettre de calculer la corrélation entre les estimateurs des moments ; la matrice de corrélation figure dans la table 4.2. Les corrélations les plus élevées sont celles qui ont trait aux paires $(\hat{\chi}_1, \hat{\chi}_{10})$, $(\hat{\chi}_{10}, \hat{\kappa})$ et $(\hat{\chi}_1, \hat{\kappa})$, par ordre de grandeur décroissant. Ce sont donc les moments qui caractérisent les deux anomalies qui font l'objet d'une estimation très corrélée : ces anomalies seraient ainsi très liées.

4.5.2 Les résultats

Nous retenons les moments présentés dans la table 4.1. Dans des explorations préliminaires, nous avons remarqué que des queues épaisses pouvaient être obtenues pour une configuration dégénérée avec très peu de transactions : par exemple, un seul ordre d'achat, aucun ordre de vente et une demande excédentaire égale à 1. Pour éviter ce type de solution, nous avons introduit un « moment » supplémentaire sous la forme d'une statistique additionnelle à reproduire : le volume relatif des transactions, noté vt et défini comme le niveau de la demande rapporté au niveau maximum de cette demande. Ce volume a été fixé à 1 %. Nous avons noté précédemment $\underline{\hat{m}}$ le vecteur (de taille p) des estimations des moments et $\underline{\theta}$ le vecteur (de taille k) des paramètres ; au cas de notre modèle, $p = 7$, $k = 4$ et ces deux vecteurs s'écrivent :

$$\underline{\hat{m}} = \begin{pmatrix} \hat{\sigma} \\ \hat{\kappa} \\ \hat{\rho}_1 \\ \hat{\rho}_{10} \\ \hat{\chi}_1 \\ \hat{\chi}_{10} \\ vt \end{pmatrix} = \begin{pmatrix} 0,0136 \\ 4,68 \\ -0,00206 \\ -0,00161 \\ 0,185 \\ 0,209 \\ 0,01 \end{pmatrix} \quad \text{et} \quad \underline{\theta} = \begin{pmatrix} \beta_m \\ \beta_v \\ \beta_i \\ \lambda \end{pmatrix} \quad (4.34)$$

Notre modèle, *a priori*, est donc identifiable puisque le nombre de paramètres est inférieur au nombre de moments. Il nous faut maintenant rapidement expliquer comment nous simulons le modèle.

Nous avons développé en C++ le modèle afin d'exploiter l'ordinateur de la façon la plus efficace possible. Il nous a semblé nécessaire de prendre un nombre suffisant d'individus pour permettre à la loi faible des grands nombres de s'exercer de sorte qu'il ne soit pas immédiat d'obtenir une distribution leptokurtique. La taille du côté du tore, notée L , a ainsi été fixée à 64 ; le nombre d'agents, noté N , est en conséquence égal à 4096. Nous supposons que la simulation se déroule sur 250 jours ; les moments simulés sont ainsi calculés sur 250 observations, nous avons noté T ce paramètre. Nous considérons en outre une période de pré-chauffage de 50 jours de sorte à moins dépendre des conditions initiales. Enfin, nous répliquons la simulation 100 fois afin de limiter au maximum l'aléa inhérent à la simulation. En effet, nous nous sommes rendus compte que cet aléa, s'il n'est pas suffisamment contrôlé, perturbe fortement la recherche du jeu de paramètres qui minimise le critère de la méthode de moments simulés.

Dans un premier temps, nous nous contentions de répliquer un petit nombre de fois la simulation et nous mettions en œuvre un algorithme numérique de résolution qui ne nécessite pas le calcul des dérivées. Plus précisément, nous utilisons, comme dans GOUDET, KANT et BALLOT (2017), l'algorithme CMA-ES et nous n'avions pas pris la mesure de la grande sensibilité des résultats par rapport à l'aléa inhérent à la simulation. En répliquant suffisamment la simulation, les résultats sont toutefois beaucoup plus stables et il est alors possible de recourir aux algorithmes éprouvés de résolution numérique. Nous avons finalement utilisé l'algorithme de LEVENBERG et MARQUARDT⁷ en calculant numériquement les dérivées. Cet algorithme est notamment disponible dans la bibliothèque *Eigen* d'algèbre linéaire que nous utilisons pour le calcul des moments, voir GUENNE-

7. Cet algorithme est une combinaison de l'algorithme de GAUSS-NEWTON et de l'algorithme de la plus forte pente ; il cherche à tirer partie de la vitesse de convergence élevée du premier et de la robustesse du second. Enfin, il présente l'avantage de ne pas nécessiter le calcul de la matrice hessienne.

TABLE 4.3 : Matrice W de la métrique statistiquement optimale

	$\hat{\sigma}$	$\hat{\kappa}$	$\hat{\rho}_1$	$\hat{\rho}_{10}$	$\hat{\chi}_1$	$\hat{\chi}_{10}$
$\hat{\sigma}$	$2,00 \times 10^6$					
$\hat{\kappa}$	346	2,23				
$\hat{\rho}_1$	8790	65,2	7060			
$\hat{\rho}_{10}$	13 700	-32,7	-1970	4820		
$\hat{\chi}_1$	-5220	-26,4	150	-221	3850	
$\hat{\chi}_{10}$	-23 000	-23,9	-406	369	-758	1220

Voir les notes du tableau 4.1.

Dans la méthode des moments simulés, il est préférable *a priori* d'utiliser la métrique qui donne, au moins asymptotiquement, les estimateurs les plus précis. Il faudrait ainsi prendre $W = \hat{\Omega}^{-1}$ où $\hat{\Omega}$ est une estimation de la matrice de variance-covariance des estimateurs des moments. La matrice W figure dans la table 4.3, telle que nous l'avons obtenue par *moving block bootstrap*. On voit alors que les poids accordés aux différents moments sont très différents : il y a par exemple un facteur 1×10^6 entre le poids de σ (l'écart-type des rendements) et le poids de κ (le kurtosis des rendements). Nous reviendrons plus loin sur les raisons d'une telle disparité. Nous délaissions dans un premier temps la métrique statistiquement optimale qui conduirait notamment à ne donner qu'un poids négligeable aux moments qui caractérisent les anomalies que nous voulons expliquer. Nous nous contentons de la métrique euclidienne, en prenant $W = I_p$ où I_p est la matrice identité d'ordre p . Le critère à minimiser est ainsi

$$\begin{aligned}
 & [\hat{\sigma}(\underline{\theta}) - 0,0136]^2 + [\hat{\kappa}(\underline{\theta}) - 4,68]^2 + [\hat{\rho}_1(\underline{\theta}) - -0,00206]^2 + [\hat{\rho}_{10}(\underline{\theta}) - -0,00161]^2 + \\
 & \quad [\hat{\chi}_1(\underline{\theta}) - 0,185]^2 + [\hat{\chi}_{10}(\underline{\theta}) - 0,209]^2 + [\hat{v}t(\underline{\theta}) - 0,01]^2
 \end{aligned}
 \tag{4.35}$$

Pour disposer des caractéristiques de la distribution des estimateurs, nous mettons en œuvre une méthode de *bootstrap* en réutilisant la tactique du *moving block bootstrap* précédemment utilisée pour calculer la matrice de variance-covariance des estimateurs

des moments. Les résultats obtenus avec la métrique euclidienne sont portés dans la table 4.4. Dans la première colonne, on trouve les résultats qui résultent du critère donné par l'équation (4.35). Ensuite nous itérons la procédure 99 fois en tirant un nouveau jeu de valeurs pour les moments à reproduire, par *moving block bootstrap*. Plus formellement, l'indice de la réplication du *moving block bootstrap* étant repéré par j , soit \hat{m}^j le vecteur des moments estimés obtenus à la réplication j . L'estimation, notée $\hat{\theta}^j$ pour la réplication j , résulte alors de la minimisation du critère suivant :

$$\hat{\theta}^j = \arg \min_{\theta} [\tilde{m}(\theta) - \hat{m}^j]' [\tilde{m}(\theta) - \hat{m}^j] \quad (4.36)$$

Nous avons programmé à l'aide de la bibliothèque standard du C++ un *pool* de tâches de sorte à employer toutes les ressources de l'ordinateur. Le micro-ordinateur que nous utilisons étant constitué de quatre *dual cores*, huit tâches peuvent s'exécuter en parallèle. Nous recherchons d'abord l'estimation qui utilise les moments calculés sur l'ensemble de l'échantillon ; cette estimation est notée $\hat{\theta}^0$. Ensuite, nous mettons en file d'attente 99 programmes qui sont pris en charge par le *pool* de tâches : quand un programme s'achève, le *pool* se charge de démarrer le programme suivant dans la file d'attente⁸. Nous utilisons $\hat{\theta}^0$ comme point de départ pour l'algorithme de LEVENBERG et MARQUARDT ; nous obtenons ainsi $\hat{\theta}^1, \dots, \hat{\theta}^{99}$. Aussi disposons-nous d'un ensemble de 100 estimations qui constitue un échantillon de l'estimateur mis en œuvre. Il nous est difficile d'aller au delà de 100 réplifications, les temps de calcul étant très longs. Dans la table 4.4, les cinq dernières colonnes décrivent la distribution empirique obtenue : moyenne, 5e percentile, 95e percentile, écart-type et coefficient de variation. Nous sommes ainsi en mesure d'évaluer la précision de nos estimations, voire de tester la significativité des paramètres du modèle.

Avec la métrique euclidienne, les résultats sont particulièrement satisfaisants. On observe notamment que les valeurs estimées pour les quatre paramètres sont toutes très

8. Nous voulions en premier lieu pouvoir nous adapter au modèle d'ordinateur utilisé. En second lieu, comme l'algorithme de résolution est itératif et s'achève quand un point fixe est obtenu, le temps de calcul d'un programme n'est pas fixé à l'avance ; il faut donc pouvoir ordonnancer les programmes.

TABLE 4.4 : Les résultats avec la métrique euclidienne

Coefficients	Estimation	Moyenne	5e perc.	95e perc.	Écart-type	Coeff. variation
β_m	1,73	1,79	1,58	2,15	0,223	0,125
β_v	1,05	1,06	1,05	1,09	0,0172	0,0162
β_i	0,0651	0,0606	0,0430	0,0722	0,008 95	0,148
λ	1600	1440	1080	1670	182	0,127

Estimation par la méthode des moments simulés. La colonne intitulée « Estimation » donne les estimations calculées à partir des moments estimés sur l'ensemble de l'échantillon. Les colonnes intitulées « Moyenne », « 5e perc. », « 95e perc. », « Écart-type » et « Coeff. variation » donnent les caractéristiques des distributions des estimateurs obtenues par *moving block bootstrap* en prenant 100 réplifications.

vraisemblables. La part de la composante idiosyncratique est faible par rapport à la part de la composante qui s'identifie à la formation mimétique des opinions. Le paramètre qui retrace la composante macro-économique joue lui-aussi un rôle important mais son estimation est un peu imprécise. Enfin le paramètre λ , qui est principalement un paramètre d'échelle dont la valeur n'a pas d'interprétation immédiate, prend une valeur raisonnable. Les quatre estimations sont toutes très significativement différentes de 0. Quand on examine les coefficients de variation, on voit que c'est le paramètre β_v qui est estimé, relativement, le plus précisément.

TABLE 4.5 : Moments simulés en fonction de la métrique utilisée

Moments	Cible	Métrique		
		euclidienne	<i>ad hoc</i>	statistiquement optimale
σ	0,0136	0,0394	0,0395	0,0115
κ	4,68	4,70	4,72	0,162
ρ_1	-0,002 06	0,000 392	-0,002 50	-0,005 62
ρ_{10}	-0,001 61	-0,003 18	-0,001 26	-0,004 73
χ_1	0,185	0,417	0,417	0,109
χ_{10}	0,209	-0,0208	-0,0200	-0,004 69
vt	0,01	0,0472	0,0476	0,0357

Voir la note de la table 4.4.

Dans la table 4.5, on trouve les valeurs des moments simulés en fonction de la métrique utilisée. Le recours à la métrique euclidienne permet de très bien reproduire l'hypothèse de marche au hasard au premier ordre (ρ_1 et ρ_{10} sont tous les deux vraiment très faibles, quasiment nuls) et la valeur élevée prise par le kurtosis. La volatilité par

grappe est moins bien reproduite : l'autocorrélation des carrés des rendements d'ordre 1 obtenue est trop élevée (0,417 alors que la valeur qu'il faudrait obtenir est égale à 0,185) et l'autocorrélation d'ordre 10 est trop faible (0,209 contre $-0,0208$). Enfin, la volatilité simulée est trop élevée tout comme le volume des transactions. Ces premiers résultats nous conduisent à nous interroger sur la métrique qu'il conviendrait d'utiliser.

La métrique mise en œuvre dans la méthode des moments simulés joue trois rôles assez différents. En premier lieu, elle assure l'homogénéité du critère employé. Si, par exemple, une autre unité de mesure était utilisée pour les rendements de l'indice boursier, les moments qui ne sont pas sans dimension en seraient affectés. Par exemple, si les rendements sont exprimés en pourcentage, l'écart-type est lui-aussi multiplié par 100 alors que le kurtosis est inchangé. Il faut donc prêter une attention suffisante à cette question d'homogénéité et, de ce point de vue, la métrique euclidienne est insatisfaisante.

En deuxième lieu, la métrique permet d'accorder un poids variable en fonction de la confiance que l'on peut attribuer à l'estimation d'un moment. On conçoit bien qu'il est préférable d'accorder un poids plus élevé à une information fiable qu'à une information douteuse. Ce principe s'oppose toutefois à notre ambition. Nous avons observé, par exemple, que le kurtosis est, relativement, estimé de façon imprécise. Ce n'est pas très étonnant puisqu'il s'agit d'un moment destiné en partie à caractériser des phénomènes rares. En retenant la métrique statistiquement optimale, on ne confère qu'une importance limitée aux deux anomalies que nous voulons reproduire. L'utilisation d'une telle métrique nous semble donc contestable.

En dernier lieu, la métrique permet de réduire sensiblement l'arbitraire qui porte sur le choix de moments. Le cas extrême suivant illustre ce point. Supposons que l'on ne retienne que trois moments ($p = 3$) et que les deux derniers moments soient identiques.

On a ainsi

$$\underline{m} = \begin{pmatrix} m_1 \\ m_2 \\ m_2 \end{pmatrix} \quad \hat{\underline{m}} = \begin{pmatrix} \hat{m}_1 \\ \hat{m}_2 \\ \hat{m}_2 \end{pmatrix} \quad \hat{\underline{\Omega}} = \hat{V}(\hat{\underline{m}}) = \begin{pmatrix} a & 0 & 0 \\ 0 & b & b \\ 0 & b & b \end{pmatrix} \quad (4.37)$$

où a est la variance estimée de l'estimateur de m_1 et b de m_2 . Nous avons supposé les deux estimateurs sans corrélation pour faire simple. Il n'est donc pas possible d'inverser $\hat{\underline{\Omega}}$ mais on prendrait dans ce cas la pseudo-inverse de $\hat{\underline{\Omega}}$ qui est égale à

$$\begin{pmatrix} \frac{1}{a} & 0 & 0 \\ 0 & \frac{1}{b} & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad (4.38)$$

En utilisant cette métrique, le critère est alors exactement celui qui résulterait de l'emploi des moments m_1 et m_2 . Il semble ainsi pertinent de tenir compte des corrélations entre les moments qui sont retenus de sorte à ne prendre en compte que l'information spécifique apportée par chaque moment.

Nous nous proposons ici de ne chercher à répondre qu'à la première préoccupation : assurer au critère une propriété d'homogénéité. Les moments que nous faisons intervenir dans le critère sont en effet dissemblables. À l'exception de σ , tous les moments sont sans dimension. Pour autant, les ordres de grandeur des moments sont différents : par exemple, les coefficients d'autocorrélation des rendements sont logiquement nuls alors que la valeur cible du kurtosis est de l'ordre de 4. Nous nous sommes attachés à construire une métrique adaptée que nous nommons *ad hoc* non pas pour signifier qu'elle serait arbitraire mais pour montrer qu'elle répond simplement à nos desseins. Nous fixons tout d'abord les poids qu'il faudrait donner à ρ_1 et à ρ_{10} : ces deux moments devraient être nuls et il faut donc pénaliser suffisamment l'écart du moment simulé et du moment cible dont la valeur est très faible. Nous avons retenu un poids égal à 1000 pour ces deux moments. Pour les autres moments, nous voulons spécifier des poids qui retracent les inverses de

l'ordre de grandeur des moments. Par exemple, le premier moment à reproduire est égal à 0,0136 et son inverse est de l'ordre de 73,5 ; pour le deuxième moment, on a respectivement 4,68 et 0,214 ; etc. Au total, les poids retenus, qui ont été arrondis à des valeurs focales, sont les suivants :

σ	κ	ρ_1	ρ_{10}	χ_1	χ_{10}	νt
100	0,25	1000	1000	5	5	1

Pour le volume des transactions, nous prenons un poids égal à 1 puisqu'il s'agissait seulement d'éviter des situations dégénérées

TABLE 4.6 : Les résultats avec la métrique *ad hoc*

Coefficients	Estimation	Moyenne	5e perc.	95e perc.	Écart-type	Coeff. variation
β_m	1,73	1,75	1,52	1,98	0,152	0,0872
β_v	1,05	1,06	1,04	1,08	0,0108	0,0103
β_i	0,0652	0,0645	0,0610	0,0684	0,002 80	0,0434
λ	1600	1620	1460	1930	125	0,0772

Voir la note de la table 4.4.

Les résultats, pour cette métrique *ad hoc*, sont portés dans la table 4.6. Deux choses sont frappantes. D'une part, l'estimation ponctuelle (la première colonne de la table) est quasiment la même que celle obtenue avec la métrique euclidienne : en fait, les nombres diffèrent après la troisième décimale et ils sont donc identiques dans les tables 4.4 et 4.6 à l'exception de l'estimation de β_i , où les résultats sont donnés avec trois chiffres. D'autre part, le gain en efficacité de l'estimateur est important. Les écarts-types sont divisés d'un facteur compris entre 1,46 (pour λ , où le gain est assez limité) et 3,19 (pour β_i). On voit aussi que les coefficients de variation sont beaucoup plus faibles et moins dispersés mais que le paramètre β_v (qui quantifie l'effet du mimétisme) reste le paramètre estimé le plus précisément.

Enfin, la métrique *ad hoc* permet de décorréler les estimateurs. Dans la table 4.2, nous avons porté la matrice des corrélations des estimateurs pour les métriques eucli-

TABLE 4.7 : Corrélation entre les estimateurs

	Métrique euclidienne				Métrique <i>ad hoc</i>			
	$\hat{\beta}_m$	$\hat{\beta}_v$	$\hat{\beta}_i$	$\hat{\lambda}$	$\hat{\beta}_m$	$\hat{\beta}_v$	$\hat{\beta}_i$	$\hat{\lambda}$
$\hat{\beta}_m$	1				1			
$\hat{\beta}_v$	0,17	1			0,040	1		
$\hat{\beta}_i$	-0,63	-0,76	1		-0,47	-0,25	1	
$\hat{\lambda}$	-0,62	-0,43	0,72	1	0,46	0,55	-0,35	1

Voir la note de la table 4.4.

dienne et *ad hoc*. Pour la première métrique, les estimateurs apparaissent très corrélés, négativement ou positivement. Cette situation s'améliore quand on utilise la métrique *ad hoc* ce qui constitue une autre forme de gains en efficacité de l'estimateur.

La métrique *ad hoc* qui donne quasiment la même estimation ponctuelle que la métrique euclidienne ne va donc pas permettre de mieux reproduire les moments par rapport à la métrique euclidienne : dans la table 4.5, on voit que les résultats sont très proches. Nous surestimons l'autocorrélation d'ordre 1 des carrés des rendements mais nous sous-estimons grandement l'autocorrélation d'ordre 10 : nous parvenons ainsi à retracer une certaine forme de volatilité par grappe mais ce n'est pas complètement satisfaisant. Dans la table 4.1, on voit aussi que la métrique statistiquement optimale donne des estimations qui reproduisent extrêmement mal par exemple le kurtosis : sa valeur simulée est proche de 0. Cette métrique n'est donc vraiment pas adaptée à notre problème.

Les perspectives de développement relèvent ainsi de deux registres. D'un côté, il nous faut réduire les temps de calcul qui restent pour le moment très élevés et qui interdisent notamment des explorations complémentaires. Initialement, nous voulions simuler le modèle en utilisant le processeur graphique de l'ordinateur ; il s'agit de l'une des motivations qui nous conduit à proposer un modèle simple. Il serait notamment possible de conduire le *bootstrap* en parallélisant la simulation du modèle sur le processeur graphique. Le contrôle de l'aléa inhérent à la simulation pourrait faire l'objet d'un réglage

plus fin. Actuellement, nous répliquons la simulation 100 fois ; ce nombre pourrait sans doute être réduit.

De l'autre côté, il nous faut mieux retracer la dynamique de la volatilité par grappe. Pour le moment, nous parvenons à la reproduire, mais nous ne reproduisons pas complètement sa signature : nous retraçons une « mémoire courte » et non une « mémoire longue ». Il reste ainsi à rechercher des spécifications plus appropriées pour l'explication qui relève de la composante macroéconomique.

4.6 Conclusion

Dans ce travail, nous proposons un *modèle basé agents* très simple, voire frustré, afin de pouvoir imputer au mimétisme des anticipations des intervenants, d'une part, la leptokurticité des rendements boursiers et, d'autre part, leur volatilité par grappe, deux anomalies contribuant à l'instabilité financière. L'indice boursier CAC 40 que nous utilisons dans ce travail présente effectivement ces deux anomalies : son kurtosis est égal à 4 alors qu'il devrait être égal à 3 si les rendements étaient distribués comme une loi normale et son autocorrélation d'ordre 1 des carrés des rendements est égal à 0,18 alors qu'elle devrait être égale à 0 si les périodes de forte volatilité n'étaient pas concentrées dans le temps.

Le modèle que nous proposons est très simple afin d'éviter deux écueils que rencontrent les *modèles basés agents* : d'une part, le fait qu'il ne soit possible ni de les estimer par une méthode économétrique ni de les valider rigoureusement par confrontation aux données ; d'autre part, le fait que ces modèles sont très souvent, sans doute par souci de réalisme, sous-identifiés. Nous nous inscrivons dans le cadre du modèle de percolation sans toutefois supposer que les agents dans chaque *cluster* puissent, à l'issue d'un processus d'échanges d'information, être tous de la même opinion. Nous préférons représenter

la diffusion des opinions sous la forme d'échanges d'information, en nombre limité, où chaque agent utilise, pour former son opinion, l'information que l'un de ses voisins lui communique.

Nous montrons qu'il est possible de mettre en œuvre une méthode de moments simulés et d'obtenir la distribution empirique des estimateurs en procédant par *moving block bootstrap*. Nous montrons aussi que l'on peut utiliser un algorithme standard de résolution numérique dès lors que l'aléa inhérent à la simulation est contrôlé en itérant un nombre de fois suffisant la simulation. Nous montrons enfin que la métrique qui est statistiquement optimale est inappropriée. En effet, les anomalies que nous voulons reproduire sont celles qui sont, en termes relatifs, caractérisées le moins précisément.

Nous obtenons des résultats qui, en premier lieu, soulignent la pertinence du modèle qui a été développé. Avec celui-ci, nous reproduisons bien les moments qui caractérisent les deux anomalies d'intérêt. Dans notre modèle, les agents utilisent trois sources d'information pour former leur opinion : une source idiosyncratique, une source relative au voisinage et une source macroéconomique. Le paramètre qui quantifie l'effet de la source relative au voisinage est celui qui est estimé le plus précisément. Nous n'infirons donc pas le rôle crucial que le mimétisme pourrait jouer pour expliquer les anomalies observées sur les marchés financiers.

Nous obtenons des résultats qui, en premier lieu, soulignent la pertinence du modèle qui a été développé. Avec celui-ci, nous reproduisons bien les moments qui caractérisent les deux anomalies d'intérêt. Dans notre modèle, les agents utilisent trois sources d'information pour former leur opinion : une source idiosyncratique, une source relative au voisinage et une source macroéconomique. Le paramètre qui quantifie l'effet de la source relative au voisinage est celui qui est estimé le plus précisément. Nous n'infirons donc pas le rôle crucial que le mimétisme pourrait jouer pour expliquer les anomalies observées sur les marchés financiers.

Conclusion générale et perspectives

Conclusion générale

Dans cette thèse, nous faisons un bref sur l'histoire du modèle *basé agent* et discutons leurs avantages et leurs limites.

Nous proposons un modèle très simple qui permet de reproduire les deux principales anomalies que l'on observe sur les marchés financiers : d'un côté, une distribution des rendements dont les queues seraient trop épaisses et, de l'autre côté, une volatilité irrégulière dans le temps du cours des actifs financiers. Nous voulons notamment montrer, en utilisant un modèle *basé agents*, qu'une formation mimétique des anticipations est en mesure de rendre compte de ces deux traits.

Les causes du scepticisme vis-à-vis du recours aux modèles *basé agents* sont discutées. Elles ne sont pas sans fondement : ces modèles sont bien souvent sur-paramétrés (dans leur ambition de décrire de manière réaliste les comportements), seulement calibrés et mal validés. Nous voulons lever ce scepticisme en proposant un modèle simple qui est estimé à l'aide d'une méthode économétrique.

Nous nous proposons de reproduire les évolutions de l'indice *CAC 40* qui sont notamment caractérisées par une distribution leptokurtique et par une volatilité par grappes.

Nous nous appuyons sur le modèle dit de percolation, développé notamment par

CONT et BOUCHAUD (2000), mais en s'écartant sensiblement des développements ultérieurs proposés, par exemple, par IORI (2002) ou FABRETTI (2013) qui cherchent, en complexifiant le modèle, à mieux reproduire les évolutions observées. Le modèle développé par IORI (2002) est seulement simulé et il reste un peu en forme de "boîte noire"; celui proposé par FABRETTI (2013) est estimé par une méthode de moments mais le modèle semble sous-identifié en raison de son très grand nombre de paramètres.

Nous paramétrons le modèle en distinguant trois sources d'information qui contribueraient à fixer les décisions d'achat ou de vente de l'actif financier : une source idiosyncratique, une source liée au voisinage de chaque agent et, enfin, une source proprement macroéconomique. La source liée au voisinage, associée au comportement mimétique, pourrait expliquer la distribution leptokurtique ; la source macroéconomique, la volatilité par grappe.

Nous discutons du critère à retenir pour estimer le modèle par la méthode des moments. Nous montrons que le critère qui repose sur la matrice de variance-covariance des moments que l'on cherche à reproduire n'est pas nécessairement le plus pertinent, même s'il semble statistiquement fondé. Par une méthode de *moving block bootstrap*, je calcule la matrice de variance-covariance des moments d'intérêt pour le cours journalier du CAC 40 depuis sa création en 1988 et nous trouvons que ce critère ne donne pas une place suffisante, par exemple, au kurtosis.

Enfin, nous présentons des premiers résultats.

Perspectives

Nous souhaiterions également confronter les résultats que nous avons obtenus avec des économistes et avec d'autres faits stylisés du marché financier. Pour le moment, nous n'avons pas calibré le modèle sous la simulation avec la méthode d'optimisation CMA-

ES (Covariance Matrix Adaptation Evolution Strategy), même nous avons simulé notre modèle avec cette méthode depuis le début. Le résultat n'est pas très favorable, il nous faudra donc bien analyser cette méthode dans le futur.

Étant donnée la durée conséquente qu'il faut pour simuler notre modèle, nous souhaitons montrer comment construire un tel modèle *basé agents* en utilisant le framework *CUDA*. Nous voudrions montrer les avantages et les limites de l'utilisation d'une unité de traitement graphique à usage général (*GPGPU*) pour développer ultérieurement notre modèle, parce qu'une disposition d'objet orientée pour concevoir les attributs des individus ne conduit pas à des accès efficaces à la mémoire. De nouvelles *GPGPU* offrent une programmabilité plus facile et maintiennent l'efficacité des calculs computationnels des *GPU* traditionnels.

Bibliographie

- ALFARANO, S., T. LUX et F. WAGNER. 2005, « Estimation of agent-based models : the case of an asymmetric herding model », *Computational Economics*, vol. 26, n° 1, p. 19–49.
- AMIHUD, Y. et H. MENDELSON. 1986, « Asset pricing and the bid-ask spread », *Journal of financial Economics*, vol. 17, n° 2, p. 223–249.
- AMIN, M. 2002, « Restructuring the electric enterprise », dans *Electricity pricing in transition*, édité par A. Faruqui et K. Eakin, Springer Science & Business Media, p. 27–50.
- ARTHUR, W. B. 1994, « Inductive reasoning and bounded rationality », *The American economic review*, vol. 84, n° 2, p. 406–411.
- ARTHUR, W. B. 2013, « Complexity economics : a different framework for economic thought », dans *Complexity Economics*, Oxford University Press.
- ARTHUR, W. B., J. H. HOLLAND, B. LEBARON, R. G. PALMER et P. TAYLER. 1996, « Asset pricing under endogenous expectations in an artificial stock market », cahier de recherche, Wisconsin Madison - Social Systems.
- AXELROD, R. 2006, « Agent-based modeling as a bridge between disciplines », dans *Handbook of computational economics*, vol. 2, édité par L. Tesfatsion et K. Judd, North Holland, p. 1565–1584.
- AXELROD, R. et W. D. HAMILTON. 1981, « The evolution of cooperation », *Science*, vol. 211, n° 4489, p. 1390–1396.

- AXELROD, R. M. 1997, *The complexity of cooperation : Agent-based models of competition and collaboration*, Princeton University Press.
- BALA, V. et S. GOYAL. 2000, « A noncooperative model of network formation », *Econometrica*, vol. 68, n° 5, p. 1181–1229.
- BALLOT, G., A. MANDEL et A. VIGNES. 2015, « Agent-based modeling and economic theory : where do we stand? », *Journal of Economic Interaction and Coordination*, vol. 10, n° 2, p. 199–220.
- BANERJEE, A. V. 1992, « A simple model of herd behavior », *The Quarterly Journal of Economics*, vol. 107, n° 3, p. 797–817.
- BANZ, R. W. 1981, « The relationship between return and market value of common stocks », *Journal of financial economics*, vol. 9, n° 1, p. 3–18.
- BARTLETT, M. S. 1946, « On the theoretical specification and sampling properties of auto-correlated time-series », *Supplement to the Journal of the Royal Statistical Society*, vol. 8, n° 1, p. 27–41.
- BASS, F. M. 1969, « A new product growth for model consumer durables », *Management Science*, vol. 15, n° 5, p. 215–227.
- BITHELL, M. et J. BRASINGTON. 2009, « Coupling agent-based models of subsistence farming with individual-based forest models and dynamic models of water distribution », *Environmental Modelling & Software*, vol. 24, n° 2, p. 173–190.
- BOOKSTABER, R. 2012, « Using agent-based models for analyzing threats to financial stability », Technical report, Office of Financial Research, US Department of the Treasury.
- BOUCHAUD, J.-P. 2008, « Economics needs a scientific revolution », *Nature*, vol. 455, n° 7217, p. 1181–1181.
- BROOKS, R. 1986, « A robust layered control system for a mobile robot », *IEEE journal on robotics and automation*, vol. 2, n° 1, p. 14–23.

- BUCHANAN, M. 2009, « Meltdown modelling », *Nature*, vol. 460, n° 7256, p. 680.
- BÜHLMANN, P. et H. R. KÜNSCH. 1999, « Block length selection in the bootstrap for time series », *Computational Statistics & Data Analysis*, vol. 31, n° 3, p. 295–310.
- CANGELOSI, A. et D. PARISI. 2012, *Simulating the evolution of language*, Springer Science & Business Media.
- CARLEY, K., J. KJAER-HANSEN, A. NEWELL et M. PRIETULA. 1992, « Plural-soar : A prolegomenon to artificial agents and organizational behavior », dans *Artificial Intelligence in Organization and Management Theory : Models of Distributed Activity*, édité par M. W. Michael Masuch, North-Holland, p. 87–118.
- CEDERMAN, L.-E. 2003, « Modeling the size of wars : from billiard balls to sandpiles », *American Political Science Review*, vol. 97, n° 1, p. 135–150.
- CHEN, S.-H., C.-L. CHANG et Y.-R. DU. 2012, « Agent-based economic models and econometrics », *The Knowledge Engineering Review*, vol. 27, n° 2, p. 187–219.
- COGNEAU, P. et V. ZAKAMOULINE. 2010, « Bootstrap methods for finance : Review and analysis », Technical report, HEC Management School.
- COLANDER, D., P. HOWITT, A. KIRMAN, A. LEIJONHUFVUD et P. MEHRLING. 2008, « Beyond DSGE models : toward an empirically based macroeconomics », *The American Economic Review*, vol. 98, n° 2, p. 236–240.
- COLLARD, M., P. COLLARD et E. STATTNER. 2012, « Mobility and information flow : percolation in a multi-agent model », *Procedia Computer Science*, vol. 10, p. 22–29.
- CONT, R. 2001, « Empirical properties of asset returns : stylized facts and statistical issues », *Quantitative Finance*, vol. 1, p. 223–236.
- CONT, R. et J.-P. BOUCHAUD. 2000, « Herd behavior and aggregate fluctuations in financial markets », *Macroeconomic dynamics*, vol. 4, n° 2, p. 170–196.

- D'AUTUME, A. 1982, « L'introduction du temps dans la théorie de l'équilibre général », *Cahiers d'Économie Politique*, vol. 7, n° 1, p. 93–127.
- DE GRAUWE, P. 2010, « Top-down versus bottom-up macroeconomics », *CESifo Economic Studies*, vol. 56, n° 4, p. 465–497.
- DING, Z., C. W. GRANGER et R. F. ENGLE. 1993, « A long memory property of stock market returns and a new model », *Journal of empirical finance*, vol. 1, n° 1, p. 83–106.
- DROGOUL, A. 1993, *De la simulation multi-agents à la résolution collective de problèmes*, thèse de doctorat, Thesis at University of Paris IV.
- EDMONDS, B. et S. MOSS. 1997, « Modelling bounded rationality using evolutionary techniques », dans *Evolutionary Computing*, édité par D. Corne et J. L. Shapiro, Springer-Verlag Berlin Heidelberg, p. 31–42.
- EFRON, B. 1992, « Bootstrap methods : another look at the jackknife », dans *Breakthroughs in statistics*, édité par S. Kotz et N. L. Johnson, Springer, p. 569–593.
- EPSTEIN, J. M. et R. AXTELL. 1996, *Growing artificial societies : social science from the bottom up*, Brookings Institution Press.
- FABRETTI, A. 2013, « On the problem of calibrating an agent based model for financial markets », *Journal of Economic Interaction and Coordination*, vol. 8, n° 2, p. 277–293.
- FARMER, J. et S. JOSHI. 2002, « The price dynamics of common trading strategies », *Journal of Economic Behavior & Organization*, vol. 49, n° 2, p. 149–171.
- FARMER, J. D. et D. FOLEY. 2009, « The economy needs agent-based modelling », *Nature*, vol. 460, n° 7256, p. 685–686.
- FERBER, J. 1995, *Les systèmes multi-agents : vers une intelligence collective*, InterEditions.
- FOGEL, D. B. 1991, « The evolution of intelligent decision making in gaming », *Cybernetics and Systems*, vol. 22, n° 2, p. 223–236.

- FRENKEN, K. 2006, « Technological innovation and complexity theory », *Economics of Innovation and New Technology*, vol. 15, n° 2, p. 137–155.
- GABORA, L. 1997, « The origin and evolution of culture and creativity », *Journal of Memetics : Evolutionary Models of Information Transmission*, vol. 1, n° 1, p. 1–28.
- GILLI, M. et P. WINKER. 2003, « A global optimization heuristic for estimating agent based models », *Computational Statistics & Data Analysis*, vol. 42, n° 3, p. 299–312.
- GYMTRASIEWICZ, P. J. et C. L. LISETTI. 2002, « Emotions and personality in agent design and modeling », *Lecture notes in computer science*, p. 21–31.
- GOUDET, O., J.-D. KANT et G. BALLOT. 2017, « Worksim : A calibrated agent-based model of the labor market accounting for workers' stocks and gross flows », *Computational Economics*, vol. 50, n° 1, p. 21–68.
- GRANGER, C. W. et Z. DING. 1996, « Varieties of long memory models », *Journal of econometrics*, vol. 73, n° 1, p. 61–77.
- GRAY, J. B. et D. W. FRENCH. 1990, « Empirical comparisons of distributional models for stock index returns », *Journal of Business Finance & Accounting*, vol. 17, n° 3, p. 451–459.
- GUENNEBAUD, G., B. JACOB *et al.*. 2010, « Eigen v3 », <http://eigen.tuxfamily.org>.
- GULTEKIN, M. N. et N. B. GULTEKIN. 1983, « Stock market seasonality : International evidence », *Journal of Financial Economics*, vol. 12, n° 4, p. 469–481.
- HALL, P., J. L. HOROWITZ et B.-Y. JING. 1995, « On blocking rules for the bootstrap with dependent data », *Biometrika*, vol. 82, n° 3, p. 561–574.
- HAMILL, L. et N. GILBERT. 2015, *Agent-based modelling in economics*, John Wiley & Sons.
- HANNEBAUER, M. 2002, *Autonomous dynamic reconfiguration in multi-agent systems : improving the quality and efficiency of collaborative problem solving*, Springer-Verlag.

- HAZY, J. K., J. GOLDSTEIN et B. B. LICHTENSTEIN. 2007, *Complex systems leadership theory : New perspectives from complexity science on social and organizational effectiveness*, ISCE Pub.
- HERLEMONT, D. 2004, « Notes sur le mimétisme et les modelisations multi-agents », cahier de recherche, YATS Finances & Technologies.
- HOLLAND, J. H. 1992, *Adaptation in natural and artificial systems : an introductory analysis with applications to biology, control, and artificial intelligence*, MIT press.
- HURST, H. E. 1951, « Long-term storage capacity of reservoirs », *Transactions of the American Society of Civil Engineers*, vol. 116, p. 770–808.
- HÄRDLE, W., J. HOROWITZ et J.-P. KREISS. 2003, « Bootstrap methods for time series », *International Statistical Review*, vol. 71, n° 2, p. 435–459.
- IORI, G. 2002, « A microsimulation of traders activity in the stock market : the role of heterogeneity, agents' interactions and trade frictions », *Journal of Economic Behavior & Organization*, vol. 49, n° 2, p. 269–285.
- JESSUA, C. 1991, *Histoire de la théorie économique*, Presses universitaires de France.
- JONDEAU, E. et M. ROCKINGER. 2003, « Testing for differences in the tails of stock-market returns », *Journal of Empirical Finance*, vol. 10, n° 5, p. 559–581.
- KIRMAN, A. P. 1983, « Communication in markets : A suggested approach », *Economics Letters*, vol. 12, n° 2, p. 101–108.
- KUNSCH, H. R. 1989, « The jackknife and the bootstrap for general stationary observations », *The annals of Statistics*, p. 1217–1241.
- KUNZ, J. C., T. R. CHRISTIANSEN, G. P. COHEN, Y. JIN et R. E. LEVITT. 1998, « The virtual design team », *Communications of the ACM*, vol. 41, n° 11, p. 84–91.

- LAIRD, J. E., A. NEWELL et P. S. ROSENBLOOM. 1987, « Soar : An architecture for general intelligence », *Artificial intelligence*, vol. 33, n° 1, p. 1–64.
- LAKONISHOK, J., A. SHLEIFER et R. W. VISHNY. 1994, « Contrarian investment, extrapolation, and risk », *The journal of finance*, vol. 49, n° 5, p. 1541–1578.
- LANE, D. A. 1993, « Artificial worlds and economics, part i », *Journal of evolutionary economics*, vol. 3, n° 2, p. 89–107.
- LEBARON, B. 2001, « A builder's guide to agent-based financial markets », *Quantitative Finance*, vol. 1, n° 2, p. 254–261.
- LEBARON, B. 2006, « Agent-based financial markets : Matching stylized facts with style », dans *Post Walrasian Macroeconomics : Beyond the Dynamic Stochastic General Equilibrium Model*, édité par D. Colander, Cambridge University Press, p. 221–236.
- LEWIS, H. R. et C. H. PAPADIMITRIOU. 1997, *Elements of the Theory of Computation*, Prentice Hall PTR.
- LEWKOWICZ, Z. et J.-D. KANT. 2008, « A multiagent simulation of a stylized french labor market : Emergences at the micro level », *Advances in Complex Systems*, vol. 11, n° 02, p. 217–230.
- LIU, J., T. DIETZ, S. R. CARPENTER, M. ALBERTI, C. FOLKE, E. MORAN, A. N. PELL, P. DEADMAN, T. KRATZ, J. LUBCHENCO *et al.*. 2007, « Complexity of coupled human and natural systems », *Science*, vol. 317, n° 5844, p. 1513–1516.
- LOUIE, M. A., K. M. CARLEY, L. HAGHSHEENASS, J. C. KUNZ, R. E. LEVITT *et al.*. 2003, « Model comparisons : docking orgahead and simvision », dans *North American association for computational social and organizational science (NAACSOS) conference*, p. 22–25.
- LUNA, F. et B. STEFANSSON. 2012, *Economic Simulations in Swarm : Agent-based modelling and object oriented programming*, vol. 14, Springer Science & Business Media.

- MALKIEL, B. G. et E. F. FAMA. 1970, « Efficient capital markets : A review of theory and empirical work », *The Journal of Finance*, vol. 25, n° 2, p. 383–417.
- MANDELBROT, B. 1963, « The variation of certain speculative prices », *The Journal of Business*, vol. 36, n° 4, p. 394–419.
- NEWELL, A. 1994, *Unified theories of cognition*, Harvard University Press.
- NÊME, C. 2001, *La pensée économique contemporaine depuis Keynes*, Economica.
- OMURTAG, A., B. W. KNIGHT et L. SIROVICH. 2000, « On the simulation of large populations of neurons », *Journal of computational neuroscience*, vol. 8, n° 1, p. 51–63.
- REINGANUM, M. R. 1981, « Misspecification of capital asset pricing : Empirical anomalies based on earnings' yields and market values », *Journal of financial Economics*, vol. 9, n° 1, p. 19–46.
- ROZEFF, M. S. et W. R. KINNEY. 1976, « Capital market seasonality : The case of stock returns », *Journal of financial economics*, vol. 3, n° 4, p. 379–402.
- SANDER, R., D. SCHREIBER et J. DOHERTY. 2000, « Empirically testing a computational model : The example of housing segregation », dans *Proceedings of the workshop on simulation of social agents : Architectures and institutions*, p. 108–115.
- SCHLÜTER, M. et C. PAHL-WOSTL. 2007, « Mechanisms of resilience in common-pool resource management systems : an agent-based model of water use in a river basin », *Ecology and Society*, vol. 12, n° 2.
- SEN, A. 1970, « The impossibility of a paretian liberal », *Journal of political economy*, vol. 78, n° 1, p. 152–157.
- SIEGEL, J. 1998, « Risque et rendement : les deux piliers des marchés », *Les Echos*, vol. 13.
- SINGH, K. 1981, « On the asymptotic accuracy of efron's bootstrap », *The Annals of Statistics*, p. 1187–1195.

- SOROS, G. 1994, *The theory of reflexivity*, Soros Fund Management.
- TANAKA, H. 2002, « A percolation model of stock price fluctuations », *Mathematical economics*, vol. 1264, p. 203–218.
- TAYLOR, S. J. 2008, *Modelling financial time series*, World scientific.
- TESFATSION, L. 2006, « Agent-based computational economics : A constructive approach to economic theory », dans *Handbook of Computational Economics*, vol. 2, édité par L. Tesfatsion et K. Judd, North Holland, p. 831–880.
- VOGEL, R. M. et A. L. SHALLCROSS. 1996, « The moving blocks bootstrap versus parametric time series models », *Water Resources Research*, vol. 32, n° 6, p. 1875–1882.
- WACHTEL, S. B. 1942, « Certain observations on seasonal movements in stock prices », *The journal of business of the University of Chicago*, vol. 15, n° 2, p. 184–193.
- WALTER, C. 2006, « Les martingales sur les marchés financiers », *Revue de Synthèse*, vol. 127, n° 2, p. 379.
- WATTS, D. J. 1999, *Small worlds : the dynamics of networks between order and randomness*, Princeton university press.
- WINKER, P., M. GILLI et V. JELESKOVIC. 2007, « An objective function for simulation based inference on exchange rate data », *Journal of Economic Interaction and Coordination*, vol. 2, n° 2, p. 125–145.
- WOLFRAM, S. 1984, « Computation theory of cellular automata », *Communications in mathematical physics*, vol. 96, n° 1, p. 15–57.
- WOLFRAM, S. 2002, *A new kind of science*, vol. 5, Wolfram media Champaign.
- ZEPPINI, P., K. FRENKEN, L. R. IZQUIERDO *et al.*. 2013, « Innovation diffusion in networks : the microeconomics of percolation », cahier de recherche, Eindhoven Center for Innovation Studies.

ZHOU, Z., W. K. V. CHAN et J. H. CHOW. 2007, « Agent-based simulation of electricity markets : a survey of tools », *Artificial Intelligence Review*, vol. 28, n° 4, p. 305–342.

Annexe A

Programmes

A.1 Calculs des statistiques du CAC 40

```
#include <iostream>
#include <fstream>
#include <vector>
#include <string>
#include <numeric>
#include <cmath>
#include <algorithm>

// Cette macro affiche le nom de son argument puis sa valeur sur le flux « cout ».
#define SHOW(arg) std::cout << #arg "\t" << (arg) << std::endl ;
#define ALL_RANGE(cont) cont.begin(), cont.end()

using namespace std ;

int main() {

    ifstream in("CAC40-cours-journalier-6clture.txt") ;

    vector<double> cac ;
    string date ;
    double valeur ;

    while ( in >> date >> valeur )
        cac.emplace_back(valeur) ;

    vector<double> r ;
    for ( size_t t = 1 ; t < cac.size() ; ++ t )
        r.emplace_back(log(cac[t]/cac[t-1])) ;

    const auto T = r.size() ;
    const auto rb = accumulate(ALL_RANGE(r), 0.) / T ;
```

```

SHOW(T)
SHOW(rb)

const auto min = *min_element(r.begin(), r.end());
const auto max = *max_element(r.begin(), r.end());
SHOW(min)
SHOW(max)

const auto median = r[r.size()/2+1];
SHOW(median)

vector<double> r_m_rb;
for ( const auto & r_t : r )
    r_m_rb.emplace_back(r_t - rb);

const auto sigma2 = inner_product(ALL_RANGE(r_m_rb), r_m_rb.begin(), 0.) / (T-1);
SHOW(sigma2)
const auto sigma = sqrt(sigma2);
SHOW(sigma)

const auto tmp4 =
    accumulate(ALL_RANGE(r_m_rb), 0., [] (double s, double x) { return s + x*x*x*x; });
SHOW(tmp4)

const auto kurtosis = tmp4 / sigma2 / sigma2 * T * (T+1) / (T-1) / (T-2) / (T-3) - 3.*(T-1)*(T-1)/(T-2)/(T-3);
SHOW(kurtosis)

const auto tmp2 =
    accumulate(ALL_RANGE(r_m_rb), 0., [] (double s, double x) { return s + x*x; });
SHOW(tmp2)

const auto tmp3 =
    accumulate(ALL_RANGE(r_m_rb), 0., [] (double s, double x) { return s + x*x*x; }); /
SHOW(tmp3)

const auto skewness = tmp3 / sigma2 / sigma * T/(T-1)/(T-2);
SHOW(skewness)

const double studentized_range = (*max_element(r.begin(), r.end()) - *min_element(r.begin(), r.end())) / sigma;
SHOW(studentized_range)

const auto s2 = inner_product(ALL_RANGE(r_m_rb), r_m_rb.begin(), 0.);
const auto rho_1 = inner_product(r_m_rb.begin()+1, r_m_rb.end(), r_m_rb.begin(), 0.) / s2;
const auto rho_10 = inner_product(r_m_rb.begin()+10, r_m_rb.end(), r_m_rb.begin(), 0.) / s2;
SHOW(rho_1)
SHOW(rho_10)

auto absr_m_rb = r;
for_each( ALL_RANGE(absr_m_rb), [] (double & r_m_rb) { if ( r_m_rb < 0 ) r_m_rb = - r_m_rb; } );
const auto average_absr_m_rb = accumulate(ALL_RANGE(absr_m_rb), 0.) / T;
for_each( ALL_RANGE(absr_m_rb), [=] (double & r_m_rb) { r_m_rb -= average_absr_m_rb; } );
const auto s2_absr_m_rb = inner_product(ALL_RANGE(absr_m_rb), absr_m_rb.begin(), 0.);
const auto nu_1 = inner_product(absr_m_rb.begin()+1, absr_m_rb.end(), absr_m_rb.begin(), 0.) / s2_absr_m_rb;
const auto nu_10 = inner_product(absr_m_rb.begin()+10, absr_m_rb.end(), absr_m_rb.begin(), 0.) / s2_absr_m_rb;
SHOW(nu_1)
SHOW(nu_10)

```

```

    auto x2 = r ;
    for_each( ALL_RANGE(x2), [] (double & r_m_rb) { r_m_rb *= r_m_rb ; } ) ;
    const auto average_x2 = accumulate(ALL_RANGE(x2), 0.) / T ;
    for_each( ALL_RANGE(x2), [=] (double & x) { x -= average_x2 ; } ) ;
    const auto s2_x2 = inner_product(ALL_RANGE(x2), x2.begin(), 0.) ;
    const auto chi_1 = inner_product(x2.begin()+1, x2.end(), x2.begin(), 0.) / s2_x2 ;
    const auto chi_10 = inner_product(x2.begin()+10, x2.end(), x2.begin(), 0.) / s2_x2 ;
    SHOW(chi_1)
    SHOW(chi_10)

    return 0 ;
}

```

A.2 Mémoire longue

```

#include <iostream>
#include <fstream>
#include <vector>
#include <string>
#include <numeric>
#include <cmath>
#include <algorithm>

// Cette macro affiche le nom de son argument puis sa valeur sur le flux « cout ».
#define DEBUG(arg) std::cout << #arg "\t" << (arg) << std::endl ;
#define ALL_RANGE(cont) cont.begin(), cont.end()

using namespace std ;

int main() {

    ifstream in("CAC40-cours-journalier-öclture.txt") ;

    vector<double> cac ;
    string date ;
    double valeur ;

    while ( in >> date >> valeur )
        cac.emplace_back(valeur) ;

    vector<double> r ;
    for ( size_t t = 1 ; t < cac.size() ; ++ t )
        r.emplace_back(log(cac[t]/cac[t-1])) ;

    const auto T = r.size() ;
    const auto rb = accumulate(ALL_RANGE(r), 0.) / T ;

    const auto min = *min_element(r.begin(), r.end()) ;
    const auto max = *max_element(r.begin(), r.end()) ;

    const auto median = r[r.size()/2+1] ;

    vector<double> r_m_rb ;

```

```

for ( const auto & r_t : r )
    r_m_rb.emplace_back(r_t - rb) ;

const auto sigma2 = inner_product(ALL_RANGE(r_m_rb), r_m_rb.begin(), 0.) / (T-1) ;
const auto sigma = sqrt(sigma2);

const auto tmp4 =
    accumulate(ALL_RANGE(r_m_rb), 0., [] (double s, double x) { return s + x*x*x*x ; }) ;

const auto kurtosis = tmp4 / sigma2 / sigma2 * T *(T+1) / (T-1) / (T-2) / (T-3) - 3.*(T-1)*(T-1)/(T-2)/(T-3) ;

const auto tmp2 =
    accumulate(ALL_RANGE(r_m_rb), 0., [] (double s, double x) { return s + x*x ; }) ;

const auto tmp3 =
    accumulate(ALL_RANGE(r_m_rb), 0., [] (double s, double x) { return s + x*x*x ; }) ;

const auto skewness = tmp3 / sigma2 / sigma * T/(T-1)/(T-2) ;

const double studentized_range = (*max_element(r.begin(),r.end())-*min_element(r.begin(),r.end())) / sigma ;

const int ds0 [] = { 1, 2, 3, } ;
const double ds [] = { .125, .25, .5, .75, 1, 1.25, 1.5, 1.75, 2, 3, } ;
// const size_t ls [] = { 1, 2, 3, 4, 5, 10, 20, 40, 70, 100, } ;

// const size_t ls [] = { 1, 2, 3, 4, 5,
6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,23,23,24,25,26,27,28,29,30,31,32,33,34,35,36,
//
37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,78,79,80,81,
//82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,100, } ;

const size_t ls [] = { 1, 2, 3, 4, 5,
6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,23,23,24,25,26,27,28,29,30,31,32,33,34,35,36,
37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,78,79,80,81,
82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,100,101,102,103,104,105,106,107,108,109,110,111,112,113,114,115,116,117,118,119,120,
121,123,123,124,125,126,127,128,129,130,131,132,133,134,135,136,137,138,139,140,141,142,143,144,145,146,147,148,149,150,
151,152,153,154,155,156,157,158,159,160,161,162,163,164,165,166,167,168,169,170,171,172,173,174,175,176,177,178,179,180,181,182,183,
184,185,186,187,188,189,190,191,192,193,194,195,196,197,198,199,200,201,202,203,204,205,206,207,208,209,210,211,212,213,214,215,216,217,
218,219,220,221,222,223,224,225,226,227,228,229,230,231,232,233,234,235,236,237,238,239,240,241,242,243,244,245,246,247,248,249,250,251,
252,253,254,255,256,257,258,259,260,};

cout << "Tableau sur la série brute.\n" ;
for ( const auto & l : ls )
    cout << '\t' << l ;
cout << endl ;
for ( const auto & d : ds0 ) {
    cout << d << '\t' ;
    vector<double> rr ;
    for ( const auto & r_t : r )
        rr.emplace_back(pow(r_t, d)) ;
    const auto rr_b = accumulate(ALL_RANGE(rr), 0.) / T ;

```

```

vector<double> rrr(rr) ;
for_each( ALL_RANGE(rrr), [=](double & rrr_t) { rrr_t -= rr_b ; } ) ;
const auto s2 = inner_product(ALL_RANGE(rrr), rrr.begin(), 0.) ;
for ( const auto & l : ls ) {
    cout << inner_product(rrr.begin()+l, rrr.end(), rrr.begin(), 0.)/s2 << '\t' ;
}
cout << endl ;
}

cout << "Tableau sur la série en valeur absolue.\n" ;
for ( const auto & l : ls )
    cout << '\t' << l ;
cout << endl ;
for ( const auto & d : ds ) {
    cout << d << '\t' ;
    vector<double> rr ;
    for ( const auto & r_t : r )
        rr.emplace_back(pow(abs(r_t), d)) ;
    const auto rr_b = accumulate(ALL_RANGE(rr), 0.) / T ;
    vector<double> rrr(rr) ;
    for_each( ALL_RANGE(rrr), [=](double & rrr_t) { rrr_t -= rr_b ; } ) ;
    const auto s2 = inner_product(ALL_RANGE(rrr), rrr.begin(), 0.) ;
    for ( const auto & l : ls ) {
        cout << inner_product(rrr.begin()+l, rrr.end(), rrr.begin(), 0.)/s2 << '\t' ;
    }
    cout << endl ;
}

// Calcul de
for ( const auto & d : ds ) {
    cout << d << '\t' ;
    vector<double> rr ;
    for ( const auto & r_t : r )
        rr.emplace_back(pow(abs(r_t), d)) ;
    const auto rr_b = accumulate(ALL_RANGE(rr), 0.) / T ;
    vector<double> rrr(rr) ;
    for_each( ALL_RANGE(rrr), [=](double & rrr_t) { rrr_t -= rr_b ; } ) ;
    const auto s2 = inner_product(ALL_RANGE(rrr), rrr.begin(), 0.) ;
    size_t tau_star = 0 ;
    for ( size_t lag = 1 ; lag < rrr.size() ; ++ lag ) {
        if ( inner_product(rrr.begin()+lag, rrr.end(), rrr.begin(), 0.) <= 0 ) {
            tau_star = lag ;
            break ;
        }
    }
    cout << tau_star << endl ;
}

return 0 ;
}

```

A.3 Nombre de cluster

```

#include <iostream>
#include "ace.hpp"

```

```

#include <utility>
#include <algorithm>
#include <numeric>
#include <cmath>
#include <map>

using namespace std ;

#include "net-set.hpp"

int main() {

    const UInt L = 10 ; // Lattice size
    const UInt N = L*L ; // Number of agents.

    for ( UInt p = 5 ; p <= 95 ; ++ p ) {
        SHOW(p)
        const vector<UInt> cluster_number = build_cluster(L, nets[p]) ;
        SHOW(cluster_number[0])
        SHOW(cluster_number[1])
        SHOW(cluster_number[N-1])
        SHOW(*max_element(RANGE(cluster_number)))
        if ( p == 10 ) return 0 ;
    }

    return 0 ;
}

```

A.4 Estimation de la variance de l'estimateur de ρ_1 pour un *AR1* par simulation et par bootstrap

```

#include <iostream>
#include <vector>
#include <random>
#include <cassert>
#include <algorithm>

#define DEBUG(arg) std::cout << #arg "\t" << (arg) << std::endl ;

using namespace std ;

vector<double> one_draw_from_moving_block_bootstrap(
    mt19937 & rng
    , vector<double> x
    , size_t block_size
) {
    assert( x.size() > block_size ) ;
    uniform_int_distribution<size_t> index(0, x.size()-block_size) ;
    const size_t blocks_nr = static_cast<double>(x.size()) / block_size + .5 ;
    vector<double> tmp ; tmp.reserve(blocks_nr*block_size) ;
    for ( size_t i = 0 ; i < blocks_nr ; ++ i ) {
        const auto start = index(rng) ;

```

```

        copy(x.begin()+start, x.begin()+start+block_size, back_inserter(tmp)) ;
    }
    return tmp ;
}

int main() {

    mt19937 rng ;
    normal_distribution<float> random_normal ;
    constexpr size_t T = 1000 ;
    constexpr double rho = .5 ;
    constexpr double mu = 1 ;

    // Estimation de la variance de l'estimateur de  $\rho_1$  pour un ARI par simulation.
    {
        constexpr size_t R = 100000 ;
        float rho_1[R] ;
        for ( size_t r = 0 ; r < R ; ++ r ) {
            double x[T] ;
            double x_m1 = mu / (1. - rho) ;
            for ( size_t t = 0 ; t < T ; ++ t ) {
                const auto xx = rho*x_m1 + mu + random_normal(rng) ;
                x[t] = xx ;
                x_m1 = xx ;
            }
            const auto xb = accumulate(x, x+T, 0.) / T ;
            for_each(x, x+T, [=](double & x) { x -= xb ; }) ;
            rho_1[r] = inner_product(x+1, x+T, x, 0.) / inner_product(x, x+T, x, 0.) ;
        }
        const auto m = accumulate(rho_1, rho_1+R, 0.) / R ;
        const auto s2 = inner_product(rho_1, rho_1+R, rho_1, 0.) / R ;
        cout << m << endl ;
        cout << s2 - m*m << endl ;
        cout << 1./T << endl ;
    }

    // Estimation de la variance de l'estimateur de  $\rho_1$  pour un ARI par bootstrap.
    {
        vector<double> x(T) ;
        {
            double x_m1 = mu / (1. - rho) ;
            for ( size_t t = 0 ; t < T ; ++ t ) {
                const auto xx = rho*x_m1 + mu + random_normal(rng) ;
                x[t] = xx ;
                x_m1 = xx ;
            }
        }
        constexpr size_t R = 100000 ;
        DEBUG(rho)
        DEBUG(1./T)
        for ( size_t number = 2 ; number <= 10 ; ++ number ) {
            float rho_1[R] ;
            for ( size_t r = 0 ; r < R ; ++ r ) {
                auto xx = one_draw_from_moving_block_bootstrap(rng, x, T/number) ;
                const auto xxb = accumulate(xx.begin(), xx.end(), 0.) / T ;
                for_each(xx.begin(), xx.end(), [=](double & x) { x -= xxb ; }) ;
                rho_1[r] = inner_product(xx.begin()+1, xx.end(), xx.begin(), 0.) /
                    inner_product(xx.begin(), xx.end(), xx.begin(), 0.) ;
            }
        }
    }
}

```

```

    }
    const auto m = accumulate(rho_1, rho_1+R, 0.) / R ;
    const auto s2 = inner_product(rho_1, rho_1+R, rho_1, 0.) / R ;
    cout << number << '\t' << T/number << '\t' << m << '\t' << (s2 - m*m) << endl ;
}
}
return 0 ;
}

```

A.5 Réseau

```

#include <iostream>
#include "ace.hpp"
#include <utility>
#include <algorithm>
#include <numeric>
#include <cmath>
#include <map>

using namespace std ;

int main() {

    mt19937 rng ;
    uniform_real_distribution<double> rand_uniform ;

    const UInt L = 10 ; // Lattice size
    const UInt N = L*L ; // Number of agents.

    cout << "std : .map<UInt, std::vector<char>>, nets_\n" ;

    // Make a network pour each connecting probability.
    bool oops = false ;
    for ( UInt p100 = 5 ; p100 <= 95 ; ++ p100 ) {
        float const p = p100 / 100. ;
        const float computed = 1.02839 - 2.57318*p + 1.58727*p*p ;
        if ( oops || (computed < 1./N) )
            oops = true ;
        const float ratio = oops ? 1./N : computed ;
        const size_t number_of_clusters = ratio*N ;
        //~ SHOW(p)
        //~ SHOW(number_of_clusters)

        // Network connections, east agents then south agents.
        vector<char> connections(2*N) ;

        // Find a net with the right number of clusters.
        while ( true ) {

            // Fix the network, i.e. populate the connections vector.
            for ( auto & cell : connections )
                cell = (rand_uniform(rng) < p) ;

            // For each agent, her cluster number, from 1 to the number of clusters.

```

```

    auto const cluster_number = build_cluster(L, connections) ;

    // Number of clusters.
    if ( *max_element(RANGE(cluster_number)) != number_of_clusters )
        continue ;

    // OK, this on is good.
    cout << '{' << p100 << ",␣{" ;
    for ( auto const & cell : connections )
        cout << (cell ? "\\1'" : "\\0'") << ",␣" ;
    cout << "},␣},\n" ;

    break ;
}
}
cout << "};\n" ;

return 0 ;
}

```

A.6 Calcul de ω et ω^{-1}

```

#include <iostream>
#include <fstream>
#include <vector>
#include <random>
#include <cassert>
#include <eigen3/Eigen/Core>
#include <eigen3/Eigen/Cholesky>

#define SHOW(arg) std::cout << #arg "\t" << (arg) << std::endl ;
#define SHOW_EIGEN(arg) std::cout << #arg "\n" << (arg) << std::endl ;
#define FOR(i, N) for ( size_t N_ = N, i = 0 ; i < N_ ; ++ i )

using namespace std ;

Eigen::VectorXd one_draw_from_moving_block_bootstrap(
    mt19937 & rng
    , Eigen::VectorXd const & x
    , size_t block_size
) {
    auto const T = x.size() ;
    assert ( T > block_size ) ;
    uniform_int_distribution<size_t> index(0, T-block_size) ;
    size_t const nr_draws = static_cast<double>(T) / block_size ;
    Eigen::VectorXd tmp(T) ;
    FOR ( i, nr_draws ) {
        const auto start = index(rng) ;
        FOR(j, block_size)
            tmp.segment(i*block_size, block_size) = x.segment(start, block_size) ;
    }
    // Last block
    const auto remain = T - (nr_draws*block_size) ;
    const auto start = index(rng) ;
    tmp.segment(nr_draws*block_size, remain) = x.segment(start, remain) ;
}

```

```

        return tmp ;
    }

Eigen::VectorXd get_returns(const string & filename) {
    ifstream in("CAC40-cours-journalier-6clture.txt") ;
    assert( in.is_open() ) ;
    vector<double> price ;
    string date ; double valeur ;
    while ( in >> date >> valeur )
        price.emplace_back(valeur) ;
    assert ( price.size() > 1 ) ;
    Eigen::VectorXd returns(price.size()-1) ;
    for ( size_t t = 0 ; t < returns.rows() ; ++ t )
        returns(t) = log(price[t+1]/price[t]) ;
    return returns ;
}

Eigen::VectorXd centered(Eigen::VectorXd const & x) { return x.array() - x.mean() ; }
// The vector x is a centered vector !
double compute_sigma(Eigen::VectorXd const & x) {
    size_t const T = x.rows() ;
    double const s2 = x.squaredNorm() ;
    return std::sqrt(s2 / (T-1)) ;
}
// The vector x is a centered vector !
double compute_kurtosis(Eigen::VectorXd const & x) {
    size_t const T = x.rows() ;
    double const s2 = x.squaredNorm() ;
    double const s4 = (x.array() * x.array()).matrix().squaredNorm() ;
    double const sigma2 = s2 / (T-1) ;
    return s4 / sigma2/sigma2 * T*(T+1) / (T-1)/(T-2)/(T-3) - 3.*(T-1)*(T-1) / (T-2)/(T-3) ;
}
// The vector x is a centered vector !
double compute_autocorrelation(Eigen::VectorXd const & x, size_t lag) {
    size_t const length = x.rows() - lag ;
    double const s2 = x.squaredNorm() ;
    return x.head(length).dot(x.tail(length)) / s2 ;
}
double compute_autocorrelation_squares(Eigen::VectorXd const & x, size_t lag) {
    Eigen::VectorXd squares = x.array() * x.array() ;
    Eigen::VectorXd centered_squares = squares.array() - squares.mean() ;
    size_t const length = x.rows() - lag ;
    double const s2 = centered_squares.squaredNorm() ;
    return centered_squares.head(length).dot(centered_squares.tail(length)) / s2 ;
}

int main() {

    mt19937 rng ;

    // Lecture des édonnes èjournalires du CAC 40.
    Eigen::VectorXd const returns = get_returns("CAC40-cours-journalier-6clture.txt") ;
    size_t const R = 100000 ;
    size_t const K = 6 ;

    Eigen::MatrixXd X(R, K) ;

    for ( size_t r = 0 ; r < R ; ++ r ) {

```

```

Eigen::VectorXd const x =
    (r == 0) ? returns : one_draw_from_moving_block_bootstrap(rng, returns, 250) ;
Eigen::VectorXd centered_x = centered(x) ;

X(r, 0) = compute_sigma(centered_x) ;
X(r, 1) = compute_kurtosis(centered_x) ;
X(r, 2) = compute_autocorrelation(centered_x, 1) ;
X(r, 3) = compute_autocorrelation(centered_x, 10) ;
X(r, 4) = compute_autocorrelation_squares(x, 1) ;
X(r, 5) = compute_autocorrelation_squares(x, 10) ;
}

Eigen::VectorXd const average = X.colwise().mean() ;
Eigen::MatrixXd const Xtilde = X.rowwise() - average.transpose() ;
Eigen::MatrixXd const Omega = (Xtilde.transpose() * Xtilde) / (R-1) ;
SHOW(returns.size())
SHOW(R)
SHOW(K)
cout << "sigma\tkurtosis\t_1\t_10\t_1\t_10\n" ;
SHOW_EIGEN(average)
SHOW_EIGEN(Omega)
Eigen::LDLT<Eigen::MatrixXd> ldl ;
ldl.compute(Omega) ;
const Eigen::MatrixXd Omega_m1 = ldl.solve(Eigen::MatrixXd::Identity(K, K)) ;
SHOW_EIGEN(Omega_m1)

return 0 ;
}

```

A.7 Relation entre le ω et la taille du bloc

```

#include <iostream>
#include <fstream>
#include <vector>
#include <random>
#include <cassert>
#include <algorithm>
#include <eigen3/Eigen/Core>
#include <eigen3/Eigen/Eigenvalues>

#define SHOW(arg) std::cout << #arg "\t" << (arg) << std::endl ;
#define SHOW_EIGEN(arg) std::cout << #arg "\n" << (arg) << std::endl ;
#define ALL_RANGE(cont) cont.begin(), cont.end()

using namespace std ;

int main() {

    string item ;
    ifstream in("mbb-cac40.out") ;
    size_t const K = 10 ;
    assert ( in.is_open() ) ;
    // Table est une matrice dont chaque ligne édpnd de block_size et chaque colonne
    // édpnd d'un èparamtre d'éèintrt.
    vector<vector<double>> table ;

```

```

while ( in >> item ) {
    assert ( item == "block_size" ) ;
    size_t block_size ;
    in >> block_size ;
    in >> item ;
    assert ( item == "Omega" ) ;
    cout << block_size ;
    Eigen::MatrixXd Omega(K, K) ;
    for ( size_t i = 0 ; i < K ; ++ i )
        for ( size_t j = 0 ; j < K ; ++ j )
            in >> Omega(i, j) ;

    //~ cout << endl ;
    Eigen::SelfAdjointEigenSolver<Eigen::MatrixXd> es(Omega) ;
    //~ cout << '\t' << es.eigenvalues().size() << '\n' ;
    cout << '\t' << es.eigenvalues().sum() << '\n' ;
    //~ cout << '\t' << Omega.sum() << '\n' ;
}

return 0 ;
}

```

A.8 Programme principal

```

#include <iostream>
#include <fstream>
#include <vector>
#include <queue>
#include <random>
#include <cassert>
#include <algorithm>
#include <chrono>
#include <thread>
#include <mutex>
#include <condition_variable>
#include <eigen3/Eigen/Core>
#include <eigen3/Eigen/Cholesky>
#include <unsupported/Eigen/NonLinearOptimization>
#include <unsupported/Eigen/NumericalDiff>

#define SHOW(arg) std::cout << #arg "\t" << (arg) << std::endl ;
#define SHOW_EIGEN(arg) std::cout << #arg "\n" << (arg) << std::endl ;
#define FOR(i, N) for ( size_t N_ = N, i = 0 ; i < N_ ; ++ i )

// We use 7 "moments"
// 1
// 2
// 3 _0
// 4 _10
// 5 _0
// 6 _10
// 7 the trade volume, demand level divide by population size
size_t constexpr nr_stats = 7 ;

// We compute 4 parameters

```

```

// 1_m
// 2_n
// 3_i
// 4
size_t constexpr nr_prms = 4 ;

size_t constexpr R = 100 ; // Number of replications for the bootstrap
size_t constexpr block_size = 250 ; // Size of the block for the moving block bootstrap
size_t constexpr L = 64 ; // Lattice size
size_t constexpr N = L*L ; // Number of agents.
size_t constexpr nr_exchanges = 5 ; // Number of exchanges between agents within each period
size_t constexpr begin_discard = 50 ; // Number of periods to discard at beginning
size_t constexpr T = 250 ; // Number of periods
size_t constexpr sim_R = 100 ; // Number of replications for the simulations

// A thread pool minimal library.
// First, instance the pool :
// ThreadPool thread_pool ;
// Then, run the tasks :
// thread_pool.add_task(some_function) ;
// thread_pool.add_task(std::bind(&SomeClass::some_member_function, some_object)) ;
// Only n threads are launched, the number of hardware threads.
class ThreadPool {
public :
// Constructor.
ThreadPool() {
// Get the natural number of threads.
auto const nr_threads = std::thread::hardware_concurrency() ;
// For now, the threads are not busy. We need to record the state of each thread :
// idle (i.e. running the endless loop) or busy (i.e. a task). At the end, we know
// if we need to wait for some working threads.
is_busy_.assign(nr_threads, false) ;
// Each thread starts an endless loop, waiting for work.
for ( size_t i = 0 ; i < nr_threads ; ++ i )
threads_.push_back(std::thread(&ThreadPool::main_loop, this, i)) ;
}
// Destructor.
~ ThreadPool() {
while ( true ) {
// Count the number of idle threads.
size_t const idle_count = count(begin(is_busy_), std::end(is_busy_), false) ;
// All the threads are idle.
if ( idle_count == is_busy_.size() ) {
// To avoid the message "terminate called without an active exception", we
// need to detach each thread.
for ( auto & thread_i : threads_ )
thread_i.detach() ;
return ;
}
// Waiting for the end of the tasks, sleeping 1 second.
using namespace std::chrono_literals ;
std::this_thread::sleep_for(1s) ;
}
}
// Each thread executes this main loop, waiting for work.
void main_loop(size_t i) {
while ( true ) {
// The task to run, extracted from the front of the queue.

```

```

        std::function<void()> task ;
    {
        // Get a lock to synchronize the acces to the queue.
        std::unique_lock<std::mutex> lock(mtx_) ;
        // Wait if the queue is empty.
        condition_.wait( lock, [&] { return ! queue_.empty() ; } ) ;
        task = queue_.front() ;
        queue_.pop() ;
    }
    is_busy_[i] = true ;
    task() ;
    is_busy_[i] = false ;
}
};

void add_task(std::function<void()> task_to_add) {
    {
        // Get a lock to synchronize the access to the queue.
        std::unique_lock<std::mutex> lock(mtx_) ;
        queue_.push(task_to_add) ;
    }
    // Tell the threads there is some work for them.
    condition_.notify_one() ;
}

private :
std::vector<std::thread> threads_ ;
// Do not use the over complicated vector<bool>.
std::vector<int> is_busy_ ;
std::queue<std::function<void()>> queue_ ;
mutable std::mutex mtx_ ;
std::condition_variable condition_ ;
};

// This function returns the vector x appropriately centered.
static Eigen::VectorXd centered(Eigen::VectorXd const & x) {
    return x.array() - x.mean() ;
}

// The vector x is a centered vector !
double compute_sigma(Eigen::VectorXd const & x) {
    size_t const T = x.rows() ;
    double const s2 = x.squaredNorm() ;
    return std::sqrt(s2 / (T-1)) ;
}

// The vector x is a centered vector !
static double compute_kurtosis(Eigen::VectorXd const & x) {
    size_t const T = x.rows() ;
    double const s2 = x.squaredNorm() ;
    double const s4 = (x.array() * x.array()).matrix().squaredNorm() ;
    double const sigma2 = s2 / (T-1) ;
    return s4 / sigma2/sigma2 * T*(T+1) / (T-1)/(T-2)/(T-3) - 3.*(T-1)*(T-1) / (T-2)/(T-3) ;
}

// The vector x is a centered vector !
static double compute_autocorrelation(Eigen::VectorXd const & x, size_t lag) {
    size_t const length = x.rows() - lag ;
    double const s2 = x.squaredNorm() ;
    return x.head(length).dot(x.tail(length)) / s2 ;
}

static double compute_autocorrelation_squares(Eigen::VectorXd const & x, size_t lag) {
    Eigen::VectorXd squares = x.array() * x.array() ;
    Eigen::VectorXd centered_squares = squares.array() - squares.mean() ;
}

```

```

    size_t const length = x.rows() - lag ;
    double const s2 = centered_squares.squaredNorm() ;
    return centered_squares.head(length).dot(centered_squares.tail(length)) / s2 ;
}

// This function returns the returns from the prices read into the file.
static Eigen::VectorXd get_returns(const std::string & filename) {
    std::ifstream in("CAC40-cours-journalier-6clture.txt") ;
    assert ( in.is_open() ) ;
    std::vector<double> price ;
    std::string date ; double valeur ;
    while ( in >> date >> valeur )
        price.emplace_back(valeur) ;
    assert ( price.size() > 1 ) ;
    size_t const T = price.size() - 1 ;
    Eigen::VectorXd returns(T) ;
    for ( size_t t = 0 ; t < T ; ++ t )
        returns(t) = log(price[t+1]/price[t]) ;
    return returns ;
}

// This function returns the agent number at (i, j) location within the L x L lattice.
// The agents are stored by columns.
static inline size_t to_n(size_t L, size_t i, size_t j) { return i + j*L ; }

// This function returns the east neighbor agent number.
static inline size_t east(size_t L, size_t n) {
    if ( n >= L*L ) return 0 ;
    const size_t j = n / L ;
    const size_t i = n - j*L ;
    const size_t m = i + (j+1)*L ;
    return (m >= L*L) ? m - L*L : m ;
}

// This function returns the south neighbor agent number.
static inline size_t south(size_t L, size_t n) {
    if ( n >= L*L ) return 0 ;
    const size_t j = n / L ;
    const size_t i = n - j*L ;
    return (i+1 == L) ? j*L : (i+1) + j*L ;
}

// This function returns the west neighbor agent number.
static inline size_t west(size_t L, size_t n) {
    if ( n >= L*L ) return 0 ;
    const size_t j = n / L ;
    const size_t i = n - j*L ;
    return (j == 0) ? i + (L-1)*L : i + (j-1)*L ;
}

// This function returns the north neighbor agent number.
static inline size_t north(size_t L, size_t n) {
    if ( n >= L*L ) return 0 ;
    const size_t j = n / L ;
    const size_t i = n - j*L ;
    return (i == 0) ? L-1 + j*L : i-1 + j*L ;
}

// This function returns a bootstrapped vector.
static Eigen::VectorXd one_draw_from_moving_block_bootstrap(
    std::mt19937 & rng
    , Eigen::VectorXd const & x

```

```

, size_t block_size
) {
    size_t const T = x.size() ;
    assert ( T > block_size ) ;
    std::uniform_int_distribution<size_t> index(0, T-block_size) ;
    size_t const nr_draws = static_cast<double>(T) / block_size ;
    Eigen::VectorXd tmp(T) ;
    FOR(i, nr_draws) {
        const auto start = index(rng) ;
        FOR(j, block_size)
            tmp.segment(i*block_size, block_size) = x.segment(start, block_size) ;
    }
    // Last block
    const auto remain = T - (nr_draws*block_size) ;
    const auto start = index(rng) ;
    tmp.segment(nr_draws*block_size, remain) = x.segment(start, remain) ;
    return tmp ;
}

// Generic functor to use with NonLinearOptimization.
template<typename S, int NX = Eigen::Dynamic, int NY = Eigen::Dynamic>
struct BaseFunctor {
    typedef S Scalar ;
    enum { InputsAtCompileTime = NX, ValuesAtCompileTime = NY } ;
    typedef Eigen::Matrix<Scalar, InputsAtCompileTime, 1> InputType ;
    typedef Eigen::Matrix<Scalar, ValuesAtCompileTime, 1> ValueType ;
    typedef Eigen::Matrix<Scalar, ValuesAtCompileTime, InputsAtCompileTime> JacobianType ;

    int inputs_, values_ ;

    // Default constructor : use dynamic sizes.
    BaseFunctor() : inputs_(InputsAtCompileTime), values_(ValuesAtCompileTime) {}
    // Constructor to specify the number of parameters et the number of values.
    BaseFunctor(int inputs, int values) : inputs_(inputs), values_(values) {}

    int inputs() const { return inputs_ ; }
    int values() const { return values_ ; }
} ;

// The used functor : the scalar type is the double type and the sizes are dynamic.
struct Functor : BaseFunctor<double> {
    // Print the results.
    bool print_ ;
    // Statistics to reproduce.
    Eigen::VectorXd stats_ ;
    // Constructor.
    Functor(Eigen::VectorXd const & stats, bool print = false) :
        BaseFunctor(nr_prms, nr_stats), print_(print), stats_(stats) {}
    // The functor.
    int operator () (Eigen::VectorXd const & beta, Eigen::VectorXd & fvec) const {

        double const beta_m = beta(0) ;
        double const beta_n = beta(1) ;
        double const beta_i = beta(2) ;
        double const lambda = beta(3) ;
        assert ( nr_prms == 4 ) ;

        double constexpr threshold = 1 ;

```

```

// Get a random number generator of type 'mt19937'.
std::mt19937 rng ;

// Adaptator to get eventually a normal distribution.
std::normal_distribution<double> rand_normal ;
// Adaptator to get eventually a uniform distribution.
std::uniform_real_distribution<> rand_uniform ;

Eigen::VectorXd stats_cumul = Eigen::VectorXd::Zero(nr_stats) ;

// Returns computed as  $P_t - P_{t-1}$ 
Eigen::VectorXd dp(T) ; // premature optimization...
// Idiosyncratic shock for each agent, draws at begin of each period.
std::array<double, N> shock ; // premature optimization...
// Expectations of agents.
std::array<double, N> expectations ; // premature optimization...

FOR(r, sim_R) {

    double dp_tm1 = 0 ;
    size_t trading_volume = 0 ;

    // For each period.
    FOR(t, begin_discard+T) {

        // Draw the idiosyncratic shock.
        std::generate( std::begin(shock), std::end(shock), [&] { return rand_normal(rng) ; } ) ;
        // Initialize the current expectations without the neighbor decisions.
        FOR(i, N)
            expectations[i] = pow(1+std::abs(dp_tm1), beta_m) * beta_i*shock[i] ;

        FOR(exchange, nr_exchanges) {
            FOR(i, N) {
                double const u = rand_uniform(rng) ;
                // Choose randomly a neighbor.
                size_t const j =
                    (u < .25) ? west(L, i) : (u < .5) ? south(L, i) : (u < .75) ? east(L,
                    i) : north(L, i) ;
                // Update the expectations with the neighbor expectation.
                expectations[i] =
                    pow(1+std::abs(dp_tm1), beta_m) * (beta_n*expectations[j] + beta_i*shock[i]
                    ) ;
            }
        }
        // Compute the excess demand and the trading volume.
        int excess_demand = 0 ;
        size_t local_trading_volume = 0 ;
        FOR(i, N) {
            if ( expectations[i] < - threshold )
                -- excess_demand ;
            else if ( expectations[i] > threshold )
                ++ excess_demand, ++ local_trading_volume ;
        }

        dp_tm1 = excess_demand / lambda ;

        if ( t >= begin_discard ) {

```

```

        dp(t - begin_discard) = dp_tm1 ;
        trading_volume += local_trading_volume ;
    }
}

if ( std::accumulate(dp.data(), dp.data()+T, 0.) == 0 ) {
    SHOW("None transactions...")
    return 1 ;
}

Eigen::VectorXd centered_dp = centered(dp) ;

stats_cumul(0) += compute_sigma(centered_dp) ;
stats_cumul(1) += compute_kurtosis(centered_dp) ;
stats_cumul(2) += compute_autocorrelation(centered_dp, 1) ;
stats_cumul(3) += compute_autocorrelation(centered_dp, 10) ;
stats_cumul(4) += compute_autocorrelation_squares(dp, 1) ;
stats_cumul(5) += compute_autocorrelation_squares(dp, 10) ;
stats_cumul(6) += double(trading_volume) / N / T ;
assert ( nr_stats == 7 ) ;

} // FOR(r, sim_R)

FOR (j, nr_stats)
    fvec(j) = stats_cumul(j)/sim_R - stats_(j) ;

if ( print_ ) {
    std::cout << "m\t_n\t_i\t\t\t\t_0\t_10\t_0\t_10\t\tv\tfit\n" ;
    FOR (j, nr_prms) std::cout << beta(j) << '\t' ;
    FOR (j, nr_stats) std::cout << stats_cumul[j]/sim_R << '\t' ;
    std::cout << (stats_cumul.array()/sim_R - stats_.array()).matrix().squaredNorm() ;
    std::cout << std::endl ;
}
return 0;
}
};

// This object centralizes all the information needed to run a estimation.
struct Estimation {
    // Number of the replications, i.e. number of the task.
    size_t r_ ;
    // Stats to reproduce, each estimation tries to reproduce a set of specific stats.
    Eigen::VectorXd stats_ ;
    // Starting point then solution
    Eigen::VectorXd beta_ ;
    // No default constructor
    Estimation () = delete ;
    // Only this constructor is allowed.
    Estimation (
        size_t r
        , Eigen::MatrixXd const & M
        , Eigen::VectorXd const & beta0
    ) : r_(r), stats_(M.row(r).transpose()), beta_(beta0) {
        std::cout << r << "\tconstructed..." << std::endl ;
    }
    // The task to run to get the estimation.
    void estimation_task () {
        std::cout << "Task\t" << r_ << "\tstarts..." << std::endl ;
        Functor functor(stats_) ;
    }
};

```

```

Eigen::NumericalDiff<Functor> numDiff(funcutor, .0001) ;
Eigen::LevenbergMarquardt<Eigen::NumericalDiff<Functor>, double> lm(numDiff) ;
int ret = lm.minimize(beta_) ;
std::cout << "Task_" << r_ << "_completes..." << std::endl ;
// The results are putted at the end of a file, one file per thread. The name of
// the file is build with the id of the thread.
std::ostringstream tmp ; tmp << std::this_thread::get_id() ;
std::ofstream out(tmp.str()+".txt", std::ios_base::app) ;
assert ( out.is_open() ) ;
out << r_ << '\t' << ret << '\t' ;
FOR(j, nr_stats)
    out << stats_(j) << '\t' ;
FOR(j, nr_prms)
    out << beta_(j) << '\t' ;
out << std::endl ;
}
};

using namespace std ;

int main() {

    mt19937 rng ;

    // Read the daily data of the CAC 40.
    Eigen::VectorXd returns = get_returns("CAC40-cours-journalier-6clture.txt") ;

    // First, get all the moments for the bootstrap.
    Eigen::MatrixXd M(R, nr_stats) ;

    for ( size_t r = 0 ; r < R ; ++ r ) {

        // Get one draw of the MBB. The first draw is the original data.
        Eigen::VectorXd const x =
            (r == 0) ? returns : one_draw_from_moving_block_bootstrap(rng, returns, block_size) ;
        Eigen::VectorXd centered_x = centered(x) ;

        // Compute the stats, from the MBB draw, the stats to reproduce.
        M(r, 0) = compute_sigma(centered_x) ;
        M(r, 1) = compute_kurtosis(centered_x) ;
        M(r, 2) = compute_autocorrelation(centered_x, 1) ;
        M(r, 3) = compute_autocorrelation(centered_x, 10) ;
        M(r, 4) = compute_autocorrelation_squares(x, 1) ;
        M(r, 5) = compute_autocorrelation_squares(x, 10) ;
        M(r, 6) = .01 ;
    }
    Eigen::VectorXd averages(nr_stats) ;

    // First optimization, to get a starting point for all subsequent optimizations.
    Eigen::VectorXd beta(nr_prms) ;
    beta(0) = 1.73058 ; // _m
    beta(1) = 1.04878 ; // _n
    beta(2) = 0.0651077 ; // _i
    beta(3) = 1603.94 ; //

    Eigen::VectorXd stats = M.row(0) ;
    SHOW_EIGEN(stats)

```

```

Functor functor(stats , /*print=*/true ) ;
Eigen::NumericalDiff<Functor> numDiff(functor , .0001) ;
Eigen::LevenbergMarquardt<Eigen::NumericalDiff<Functor>, double> lm(numDiff) ;
int ret = lm.minimize(beta) ;
SHOW(ret)
SHOW_EIGEN(beta)

// Build a vector of size R for each estimation to run. The 0 element is not
// used but is included for the later calculations.
vector<Estimation> estimations ; estimations.reserve(R) ;
FOR(r, R)
    estimations.emplace_back(r, M, beta) ;
{
    // Create a pool of threads, using all the resources of the computer.
    ThreadPool thread_pool ;
    // Check if work as already done.
    vector<size_t> already_done ;
    {
        ifstream in("ace-sim.txt") ;
        string line ;
        while ( getline(in, line) ) {
            already_done.emplace_back(stoi(line)) ;
        }
    }
    // Add to the queue all the work to do.
    for ( size_t r = 1 ; r < R ; ++ r )
        if ( find(begin(already_done), end(already_done), r) == end(already_done) )
            thread_pool.add_task(bind(&Estimation::estimation_task, estimations[r])) ;
} // At block exit, all tasks are completed.
// Compute for each parameter the mean, the 5th percentile, the median, and the 95th
// percentile.
FOR(j, nr_prms) {
    Eigen::VectorXd tmp(R) ;
    FOR(r, R)
        tmp(r) = estimations[r].beta_(j) ;
    sort(tmp.data(), tmp.data()+R) ;
    cout << j << '\t' << tmp.mean() << '\t' << tmp(.05*R) << '\t' <<
        tmp(.5*R) << '\t' << tmp(.95*R) << '\n' ;
}
return 0;
}

```