
Rapport mini-projet

Modélisation – Représentations et analyse des modèles

Sujet No.2

Modélisation des non-linéarités dans les évolutions des valeurs boursières

Réalisé par le groupe G :

Alexandre VALLET

Gaspard LAFONT

Tomas JOUVEN

Table des matières

1	Introduction	1
1.1	Contexte et objectif	1
1.2	Etat de l'art	1
1.3	Contribution	2
1.4	Structure du rapport	2
2	Modèle à état continu à temps discret	3
2.1	Mise en place du modèle et hypothèses	3
2.2	Mise en équation du modèle	4
3	Résultats de simulation et analyse du modèle	5
3.1	Modèle initiale	5
3.1.1	Simulation python	5
3.1.2	Analyse du modèle	5
3.2	Nouveau modèle	6
3.2.1	Rélexion	6
3.2.2	Simulation python	6
4	Analyse de l'équilibre	8
4.1	Calcule des points d'équilibres	8
4.2	Analyse de stabilité du points d'équilibre	8
4.3	Exemple de non-stabilité par simulation	9
5	Modèle à évènements discrets	10
5.1	Automate hybride	10
5.2	Simulation de l'automate hybride	11
6	Conclusion (1 page max)	12
6.1	Conclusion sur les modèles proposés	12
6.2	Perspectives	12
	Annexes	12
	Annexe 1	13

Chapitre 1

Introduction

1.1 Contexte et objectif

La modélisation consiste à représenter et analyser l'évolution d'un système à l'aide d'un modèle, qu'il soit exploitable analytiquement ou numériquement. Cette démarche est fondamentale dans la recherche scientifique et est largement utilisée dans divers domaines, y compris la finance, où plusieurs modèles, dont le célèbre "modèle Black-Scholes", ont été créés dans le but de prédire les évolutions des actions en bourse.

Il est crucial de reconnaître que prédire le rendement du marché boursier, avec la multitude de facteurs impliqués, représente l'un des défis les plus complexes. Notre projet vise à établir un modèle simple pour l'évolution des prix d'une action boursière en prenant en compte certaines non-linéarités. Cette approche cherche à offrir une compréhension claire et accessible tout en reconnaissant la complexité inhérente à la prédiction des mouvements du marché financier.

1.2 Etat de l'art

Dans l'histoire de la finance, les remises en question des modèles mathématiques et les travaux de recherche sont souvent déclenchés par des événements majeurs, tels que des crises financières, qui perturbent les cours. Ces situations incitent les universitaires à examiner de près la dynamique du marché et à développer des modèles permettant de mieux prédire les fluctuations des cours, notamment au niveau des non-linéarités du prix des actifs.

La prise en compte des non-linéarités dans l'analyse des cours boursiers revêt une importance capitale en raison de la nature intrinsèquement complexe et dynamique des marchés financiers. Ces non-linéarités émergent des interactions multiples et souvent imprévisibles entre divers facteurs qui influent sur les prix des actifs.

L'un des premiers modèles à émerger est la "Théorie de la Spéculation", élaborée par Louis Bachelier [1]. Dans ce modèle, le cours des actions est considéré comme sans mémoire, ce qui signifie que la probabilité de chaque mouvement futur dépend uniquement du prix actuel. Cependant, ce modèle présente des limites en négligeant de nombreux facteurs importants.

Un autre modèle fascinant et plus actuel utilise les mouvements Browniens [2]. Les mouvements browniens sont utilisés pour modéliser le comportement aléatoire des prix des actifs financiers sur les marchés. Ce modèle repose sur l'idée que les variations de prix d'un actif sont imprévisibles et suivent un processus stochastique continu dans le temps. Cette approche permet de simuler divers scénarios pour anticiper les fluctuations aléatoires des prix, jouant un rôle crucial dans des modèles tels que le "modèle Black-Scholes" utilisé pour évaluer les options financières [3]. C'est notamment ce dernier modèle qui est le plus utilisé et joue un rôle crucial dans l'analyse des options d'achat européennes. Les économistes Black et Sholes ont notamment reçu le prix Nobel d'économie en mettant en place ces équations :

$$C = S_0 N(d_1) - X e^{-rT} N(d_2)$$

$$d_1 = \frac{\ln(S_0/X) + (r + (\sigma^2)/2)T}{\sigma\sqrt{T}}$$

$$d_2 = d_1 - \sigma\sqrt{T}$$

L'approche Brownienne demeure malheureusement trop complexe, rendant son analyse dynamique difficile dans notre cas. La variabilité des résultats, notamment en raison de la volatilité, nécessite une attention particulière dans l'application pratique de ces modèles.

1.3 Contribution

Dans le cadre de notre projet, notre objectif principal était de simplifier et rendre modélisables les équations qui déterminent l'évolution des prix des actifs financiers sur les marchés, en utilisant les outils disponibles. Ces équations, souvent complexes avec de nombreux paramètres connus ou inconnus, nécessitent une modélisation adéquate pour accéder aux valeurs de ces paramètres. Ceci est essentiel pour choisir des stratégies d'investissement optimales.

Notre travail offre une première approche visant à comprendre comment les prix des actifs peuvent réagir aux différentes conditions du marché. La flexibilité et l'adaptabilité de notre modèle face à une variété de paramètres en font un outil puissant pour s'ajuster aux caractéristiques spécifiques de chaque actif financier ainsi qu'aux propriétés des décisions d'investissement prises.

Cette première approche pourrait, par exemple, aider à déterminer quelle stratégie d'investissement serait la plus appropriée pour un actif particulier. En fournissant une méthodologie claire et flexible, notre modèle offre une base solide pour analyser et interpréter les comportements des actifs financiers dans des conditions variées du marché, contribuant ainsi à une prise de décision plus informée dans le domaine de l'investissement.

1.4 Structure du rapport

Le Chapitre 2 présente *le modèle mathématique à état continu*. Le Chapitre 3 porte sur *l'analyse des résultats de simulation obtenus en utilisant le modèle à état continu*. Le Chapitre 4 se focalise sur *la recherche des points d'équilibres et l'analyse de leur stabilité*. Le Chapitre 5 se concentre sur *le développement du modèle à événements discrets*. Finalement, les conclusions et perspectives sont présentées. Enfin une archive des programmes Matlab ou Python est également fournie avec le rapport.

Chapitre 2

Modèle à état continu à temps discret

2.1 Mise en place du modèle et hypothèses

Pour notre problème de l'évolution de valeurs boursières, nous allons définir deux variables :

- (p) : la valeur de l'action à l'instant considéré.
- (s) : le nombre d'actions que la masse possède (par "masse", nous entendons la population impliquée dans la bourse).

Maintenant que nous avons défini les variables que nous allons étudier, il convient de fixer leurs évolutions dans le temps à travers différentes fonctions.

Posons g , la fonction qui fixe la variation de la variable p entre les instants t et $t + 1$. La valeur de l'action au jour $t + 1$ ne dépendant que des variables au jour t , cette fonction prends donc en paramètre le prix de l'action p et le nombre d'actifs en possession des acheteurs s à l'instant t :

$$p_{t+1} - p_t = g(s_t, p_t) \quad (2.1)$$

Posons f , la fonction qui fixe la variation de la variable s entre les instants t et $t + 1$. Nous allons considérer dans notre cas que les opérations des boursiers arrivent après la réévaluation de la valeur de l'actif p . Ainsi cette fonction prend en paramètre la nouvelle valeur de p (jours $t + 1$), l'ancienne valeur de s (jours t) ainsi que la variation de p entre les instants t et $t + 1$ (nous expliquerons ce choix dans le point 5)) :

$$s_{t+1} - s_t = f(s_t, p_{t+1}, p_{t+1} - p_t) \quad (2.2)$$

De part nos connaissances sur la relation entre le prix et la demande et notre sens logique, nous allons émettre des hypothèses sur les dérivées partielles de ces fonctions qui vont nous permettre de préciser notre modèle.

1) Si la demande augmente, la valeur de l'action augmente. Ainsi la variation du prix de l'actif (fonction g) à une relation croissante avec le nombre d'actions en possessions s_t , on a donc :

$$\frac{\partial g(s_t, p_t)}{\partial s_t} > 0$$

2) En présence d'un niveau de prix élevé p_t la demande diminuera, entraînant alors une réduction des prix, donc une variation négative. On a donc la relation :

$$\frac{\partial g(s_t, p_t)}{\partial p_t} < 0$$

3) Les épargnants ont tendance à diversifier leurs investissements. Ainsi si s_t augmente, les acteurs diminueront leurs transactions. On a donc :

$$\frac{\partial f(s_t, p_{t+1}, p_{t+1} - p_t)}{\partial s_t} < 0$$

4) En cas d'augmentation soudaine de la valeur de l'actif au jour $t + 1$, les actionnaires vont tirer profit des bonnes performances du marché en vendant leurs capitaux investis :

$$\frac{\partial f(s_t, p_{t+1}, p_{t+1} - p_t)}{\partial p_{t+1}} < 0$$

5) Les épargnants ont des attentes adaptatives, ce qui signifie qu'ils supposent que les tendances observées entre deux jours se maintiendront au cours de la période suivante. Ainsi, si on observe une variation positive du prix de l'action, la masse devrait acheter davantage d'actifs dans l'espoir que la croissance continue :

$$\frac{\partial f(s_t, p_{t+1}, p_{t+1} - p_t)}{\partial (p_{t+1} - p_t)} > 0$$

On remarque que les hypothèses 4) et 5) force le système dans des directions différentes. En effet, la concurrence de leurs effets a une importance cruciale dans l'analyse de notre modèle (nous y reviendrons plus tard).

2.2 Mise en équation du modèle

A partir des cinq propositions sur f et g ainsi que des hypothèses sur leurs régularités, on peut affirmer que les points d'équilibre \bar{s} et \bar{p} sont uniques, s'ils existent (voir section point d'équilibre).

On peut donc écrire notre modèle :

$$S : \begin{cases} s_{t+1} - s_t = F(s_t - \bar{s}, p_{t+1} - \bar{p}, p_{t+1} - p_t) \\ p_{t+1} - p_t = G(s_t - \bar{s}, p_t - \bar{p}) \end{cases} \quad (2.3)$$

$$(2.4)$$

Avec F et G satisfaisant les mêmes conditions que f et g , dans ce contexte on a naturellement : $F(0, 0, 0) = G(0, 0) = 0$

Avec le changement de variable suivante :

$$— S_t = s_t - \bar{s} \text{ et } P_t = p_t - \bar{p}$$

On obtient le modèle de point d'équilibre $O(0,0)$:

$$S : \begin{cases} S_{t+1} - S_t = F(S_t, P_{t+1}, P_{t+1} - P_t) \\ P_{t+1} - P_t = G(S_t, P_t) \end{cases} \quad (2.5)$$

$$(2.6)$$

Dès lors, nous pouvons préciser nos fonctions en mettant en lumière des relations **non-linéaires** :

$$S : \begin{cases} S_{t+1} - S_t = -AP_{t+1} - BS_t^3 + E(P_{t+1} - P_t) \\ P_{t+1} - P_t = CS_t - DP_t^3 \end{cases} \quad (2.7)$$

$$(2.8)$$

Où tous les coefficients $A B C D E$, dont la signification peut être facilement déduite de la discussion précédente sur les hypothèses 1 à 5, sont **réels et positifs**.

En manipulant ce système, on obtient finalement le modèle itérable (l'état $t + 1$ ne dépend que de l'état t) :

$$S : \begin{cases} S_{t+1} = (1 + ec)S_t - aP_t - bS_t^3 - edP_t^3 \\ P_{t+1} = cS_t + P_t - dP_t^3 \end{cases} \quad (2.9)$$

$$(2.10)$$

$$a = A, b = B, c = C, d = D, e = E - A, \text{ avec } e + a > 0$$

Où les constantes a, b, c, d et e représentent (sous les hypothèses) :

- **a** : L'effet inverse du prix sur l'épargne. Il représenterait la réaction des investisseurs par une forte baisse des prix, ce qui entraînerait une accélération de la baisse des prix.
- **b** : L'effet non linéaire de l'épargne sur elle-même. Ce coefficient intervient dans la façon dont le niveau des prix affecte l'offre sur le marché. Il est influencé par le comportement d'épargne des investisseurs en réponse à la volatilité du marché.
- **c** : L'effet direct de l'épargne sur le prix. Il influe sur la manière dont les choix d'épargne des individus impactent la diversification de leurs investissements.
- **d** : L'effet non linéaire des prix sur eux-mêmes. Ce coefficient intervient dans la propension des épargnants à tirer profit des bonnes performances du marché.
- **e** : L'impact de l'épargne sur le prix. Il représenterait la confiance des investisseurs, par conséquent leur volonté d'épargner ou d'investir.

En résumé, ces coefficients permettent de paramétrer le modèle, déterminant ainsi comment les différentes variables du système interagissent et évoluent au fil du temps. Ainsi, des valeurs spécifiques accordées à ces coefficients influenceront les résultats et les comportements simulés par le modèle. Les hypothèses formulées et le choix des paramètres permettent de rediriger le marché lors de ses fluctuations.

Chapitre 3

Résultats de simulation et analyse du modèle

3.1 Modèle initiale

3.1.1 Simulation python

En prenant le modèle S (2.9)(2.10), sur une période de 500 jours avec pour paramètres : $a = 1.2, b = 0.5, c = 0.4, d = 0.1$ et $e = -0.9$. On obtient les graphiques en utilisant le code de la Figure 5.1 :

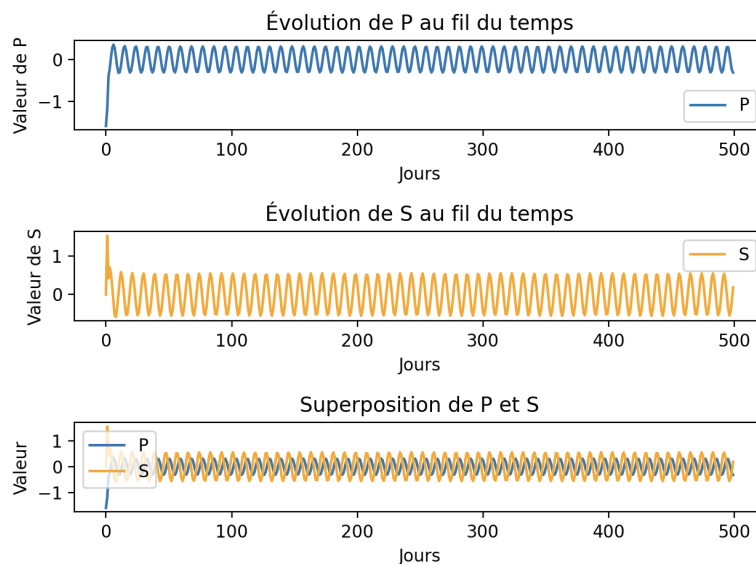


FIGURE 3.1 – Analyse du modèle donnée

3.1.2 Analyse du modèle

De ces graphiques nous pouvons tirer plusieurs conclusions. Tout d'abord, nous pouvons remarquer que le modèle tend à osciller autour du point d'équilibre 0 et reste borné par $[-1,1]$, que ce soit pour S ou pour P (ceci sera développé dans la partie dédiée).

Ces deux derniers points sont réellement importants pour l'adaptivité de notre modèle, en jouant avec les paramètres a, b, c et e nous pouvons simuler la variation de la valeur d'un actif en particulier (ex : l'or \Rightarrow confiance très haute, crypto \Rightarrow très sensible...). Mais comment ramener ce modèle borné à une prédiction réelle du cours d'une action ?

1. Les cours ont des moyennes, ou du moins une interpolation linéaire connue car la tendance reste assez constant dans le temps (ex : l'or \Rightarrow valeurs = $102.9 \times \text{Année} + 1049.41$, interpolation sur les dix dernières années), que l'on pourrait ainsi ajouter à notre modèle qui représenterait alors uniquement la variation autour de cette droite (ou de la moyenne).
2. Quant à la hauteur de ces variations, il faut considérer la volatilité σ de l'actif. La volatilité est un indicateur qui mesure l'amplitude des hausses et des baisses de l'action. La valeurs sera d'autant plus

élevée que les cours des marchés sont instables. On pourrait alors trouver une relation entre la variation $[-1,1]$ et la volatilité (ex : l'or $\Rightarrow 6.58\%$, moyenne sur les dix dernières années). Cette relation existe, mais reste très complexe à trouver, en effet nous cherchons la solution d'une équation fonctionnelle qui dépasse largement les attendues de notre projet. Pour cela nous allons simplifier le problème avec la constante α que nous fixons par l'observation des anciennes données (ex l'or : $\Rightarrow \alpha = 5$).

Nous obtenons :

$$V_t = P_I(t) * (1 + P_t * \sigma / \alpha), \text{ avec } V_t \text{ la nouvelle valeurs de l'actif et } P_I(t) \text{ l'interpolation linéaire empirique}$$

De plus, on pourrait croire que l'évolution est sinusoïdale et périodique. En diminuant l'échelle, on observe alors quelques irrégularités, mais on peut tout de même affirmer que le modèle semble converger vers une trajectoire suivant une sinusoïde. Nous pouvons aussi confirmer la validité de notre modèle. En effet, on a bien une évolution du prix et des stocks qui s'autorégule/s'auto-corrige.

Malgré tout, nous pouvons soulever un défaut de notre modèle tel qu'il est : **Les valeurs de S et de P ne dépendent que d'elles mêmes, il n'y a aucun effet extérieur.** Or on sait que les marchés financiers dépendent énormément de l'influence de la sphère géopolitique ou encore des réseaux sociaux. On peut citer notamment le cours des Crypto-monnaie qui peut s'effondrer suite à un simple tweet.

3.2 Nouveau modèle

3.2.1 Réflexion

Afin de modéliser cela, on décide d'influer sur les stocks S_{t+1} à l'aide de variables aléatoires X et Y. Posons X une variable aléatoire suivant une loi normale d'espérance nul (les cas de grande influence sont rares) et d'écart-type σ . Cette variable définira la correction journalière sur S_{t+1} . Malgré tout, on ne peut pas affirmer qu'une influence sur les actionnaires a lieu tous les jours. Pour corriger ce problème on pose Y, une variable aléatoire suivant une loi uniforme entre $[0,1]$. De cela on peut définir $\mathbb{1}_{[0,n]}(Y)$ qui, une fois multiplié à X, dictera une influence sur le marché aux jours $t + 1$ sous une probabilité égale à n (ex : $\mathbb{1}_{[0,0.4]}(Y) \Rightarrow$ une correction a lieu sous une probabilité de 0.4).

Ce nouveau modèle ajoutant des événements aléatoires nous permet de mettre en lumière la réactivité du modèle face à des imprévus et le rendre plus proche d'un cas concret.

Soient X et Y comme définit ci-dessus : $X \sim \mathcal{N}(0, 0.2^2)$, $Y \sim U([0, 1])$.

Notre système intégrant des variables aléatoires est :

$$S_{va} : \begin{cases} S_{t+1} = (1 + ec)S_t - aP_t - bS_t^3 - edP_t^3 + \mathbb{1}_{[0,0.8]}(Y) * X \\ P_{t+1} = cS_t + P_t - dP_t^3 \end{cases} \quad (3.1)$$

$$(3.2)$$

3.2.2 Simulation python

En gardant les mêmes paramètres a,...,e on obtient à l'aide du code de la Figure 5.2 les courbes :

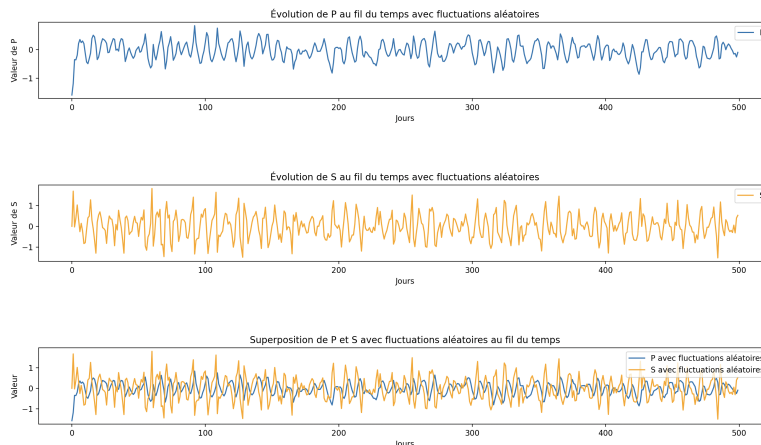


FIGURE 3.2 – Analyse du modèle avec ajout de variables aléatoires

On observe alors que les événements aléatoires simulent parfaitement un cours boursier et que le modèle continue de suivre le comportement attendu (auto-correction/régulation).

Simulons maintenant **le cours de l'or** suivant notre nouveau modèle en considérant les données empiriques de l'actif :

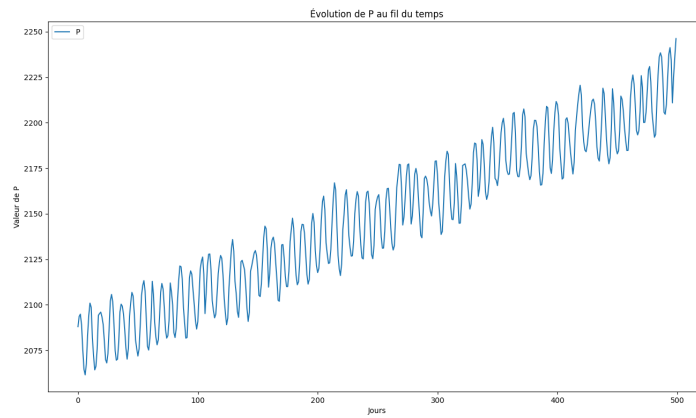


FIGURE 3.3 – Prédiction du cours de l'or dans les prochains 500 jours

On obtient un graphique cohérent. Notre modèle est donc assez proche de la réalité et pose des bases valides pour la conception de notre automate.

Chapitre 4

Analyse de l'équilibre

4.1 Calcule des points d'équilibres

Analysons la stabilité de notre modèle. Comme expliqué en section 2.2, nous sommes partie de l'hypothèse que si un point d'équilibre existe, alors il est unique. Nous avons alors centralisé nos variables autour de ce point d'équilibre pour finalement avoir un modèle oscillant autour de 0 ($S_t = s_t - \bar{s}$ et $P_t = p_t - \bar{p}$). Nous pouvons alors affirmer, par conséquence des dernières opérations, que $O(0,0)$ est l'unique point d'équilibre de notre modèle final ($F(0,0,0) = G(0,0) = 0$). L'idée derrière cette démarche est que les points d'équilibres des valeurs boursières sont connus avec les données dont on dispose, il n'est donc pas intéressant de calculer directement \bar{s} et \bar{p} , d'où la centralisation autour de 0. Nous allons par conséquent montrer les conditions de stabilité du point $O(0,0)$.

4.2 Analyse de stabilité du points d'équilibre

Pour étudier la stabilité d'un point d'équilibre, ici $O(0,0)$, il faut montrer que la norme des valeurs propres de la matrice Hessienne de notre modèle en ce point est inférieur à 1 (donc de la matrice d'évolution F du système linéarisé autour du point d'équilibre).

On calcule donc la matrice hessienne de nos deux fonctions :

$$H(s, p) = \begin{bmatrix} \frac{\partial F(s, p)}{\partial s} & \frac{\partial G(s, p)}{\partial p} \\ \frac{\partial G(s, p)}{\partial s} & \frac{\partial G(s, p)}{\partial p} \end{bmatrix} = \begin{bmatrix} 1 + ec - 3bS^2 & -a - 3edP^2 \\ c & 1 - 3dP^2 \end{bmatrix}$$

Appliqué au point d'équilibre $O(0,0)$, on trouve :

$$H(0, 0) = \begin{bmatrix} 1 + ec & -a \\ c & 1 \end{bmatrix}$$

Pour trouver les valeurs propres de la matrice H il suffit de résoudre le système suivant :

$$\text{Det}(H(0, 0) - \lambda I_2) = \text{Det} \begin{bmatrix} 1 + ec - \lambda & -a \\ c & 1 - \lambda \end{bmatrix} = 0$$

$$\text{Soit : } \lambda^2 - \lambda(2 + ec) + 1 + c(e + a) = 0$$

Après résolution d'une équation du second degré à déterminant négatif on obtient :

$$\lambda_i = \frac{2 + ec \pm i\sqrt{e^2c^2 - 4ca}}{2}$$

Un système à temps discret est asymptotiquement et exponentiellement stable si et seulement si les valeurs propres de la matrice d'évolution $F=H(0,0)$ ont toutes un module strictement inférieur à 1 : $|\lambda_i| < 1$. Soit :

$$|\lambda_i| < 1 \iff \frac{(2 + ec)^2 + e^2c^2 - 4ac}{2} < 1 \iff e^2c^2 + 2ec - 2ca < -1$$

Ainsi, nous avons défini la condition sur les paramètres de simulation afin de caractériser la stabilité du point d'équilibre.

4.3 Exemple de non-stabilité par simulation

En utilisant ce dernier critère, nous pouvons affirmer qu'avec les valeurs des paramètres utilisés jusqu'à maintenant ($a = 1.2, b = 0.5, c = 0.4, d = 0.1$ et $e = -0.9$), le point d'équilibre $O(0,0)$ est **instable**. Montrons cela par simulation.

Tout d'abord, on vérifie que $O(0,0)$ est un point d'équilibre en entrant comme valeurs initiales $S_0 = 0$ et $P_0 = 0$ dans notre programme :

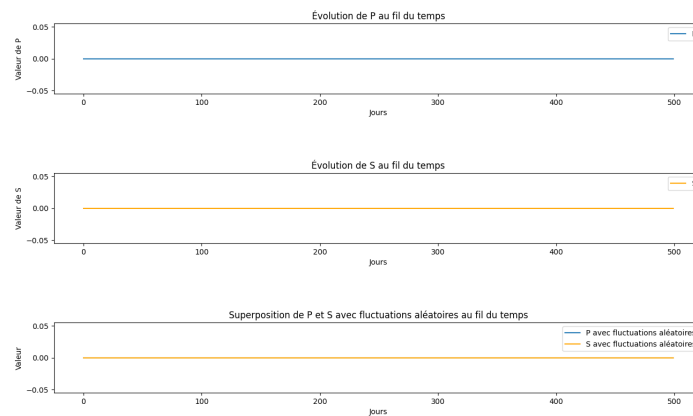


FIGURE 4.1 – Simulation avec conditions initiales à l'équilibre

On a donc bien un système constant à $(0,0)$.

Maintenant, on entre les valeurs initiales $S_0 = 10^{-7}$ et $P_0 = 10^{-7}$ (très proche du point d'équilibre) :

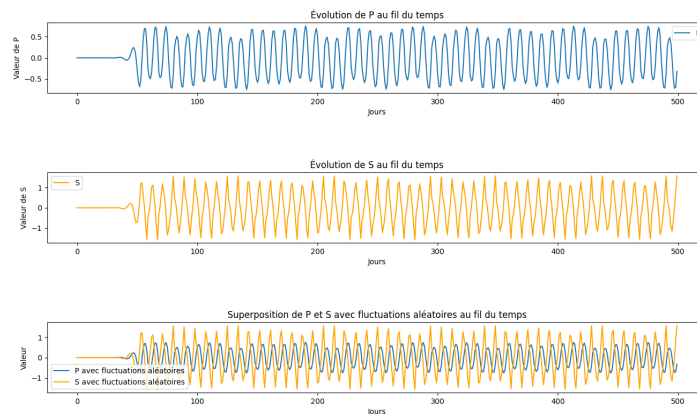


FIGURE 4.2 – Simulation de la non-stabilité du point d'équilibre

On observe alors que le système s'éloigne au fur et à mesure du point d'équilibre $O(0,0)$, avec des très faibles oscillations avant le 40ième jours qui finissent par exploser. On donc bien un système **instable** avec ces valeurs de paramètres.

Modèle à évènements discrets

L'objectif principal de cette section est de concevoir un automate permettant à une institution bancaire de gérer efficacement des opérations d'achat et de vente basées sur les fluctuations des marchés boursiers. Pour ce faire, nous partons du principe que la banque a uniquement accès aux prix courants des actions, sans connaître le volume quotidien des actions échangées.

$$\begin{cases} (1) -AP_{t+1} - BS_t^3 + E(P_{t+1} - P_t) & (5.1) \\ (2) CS_t - DP_t^3 & (5.2) \\ (3) P_{\Delta}(t+1) - P_{\Delta}(t) = \frac{1}{100} [(P_{t+1} - P_t) - (P_{t-99} - P_{t-100})] & (5.3) \end{cases}$$


$$\left(\begin{array}{l} M : \text{Argent physique que dispose l'automate} \\ A : \text{Nombre d'actions que dispose l'automate} \\ \Sigma : \text{évènement extérieur qui force la banque à arrêter le processus.} \\ Q : \text{Nombre d'actions achetées ou vendues par transaction} \\ Q = (P_{t+1} - P_t - P_{\Delta}) \times M \end{array} \right) \quad (5.4)$$

États : Notre modèle est conçu pour refléter fidèlement les variations des valeurs boursières. Dans cette optique, notre automate se compose de deux états. Le premier état représente les étapes d'analyse des données, tandis que le second état représente les étapes d'achat et de vente. Cette structure stratégique permet de concentrer toute la dynamique de l'automate sur les processus critiques d'achat et de vente.

P_Δ représente la variation moyenne des prix des actions sur les 100 derniers jours, calculée comme la moyenne des différences quotidiennes des prix des actions sur cette période. P_Δ est initialisé dans l'état q_1 , l'automate reste dans cet état jusqu'à ce que P_Δ soit correctement initialisé.

$$P_\Delta(t) = \frac{1}{100} \sum_{m=0}^{99} (P_{t-m} - P_{t-(m+1)})$$

Transitions : Les transitions au sein de l'état de transactions sont conçues pour optimiser le capital de la banque en réagissant aux fluctuations des prix des actions. L'automate exécute des achats lorsque les prix tombent sous le seuil défini par P_Δ et procède à la vente lorsque les prix dépassent ce même seuil. Cette stratégie s'appuie sur l'observation, mentionnée à la page (3) de notre modèle, selon laquelle de grandes variations de prix d'un jour à l'autre sont souvent suivies par un renversement de tendance. Cela suggère qu'une variation significative des prix pourrait être un indicateur fiable pour prévoir le mouvement futur des valeurs boursières. Pour déterminer la quantité de fonds à acheter, nous avons rendu celle-ci proportionnelle à la différence entre la variation du prix et P_Δ , c'est-à-dire proportionnelle à :

$$(P_{t+1} - P_t - P_\Delta(t))$$

Lorsque l'automate est dans l'état de transactions, il peut revenir à l'état d'analyse de données si le capital de la banque est négatif, ou si un événement extérieur, résultant des stratégies décidées par la banque, impose ce changement. Durant cette transition, la variable de temps est réinitialisée. La banque dans cette situation se retrouve donc obligé de rajouter des fonds via un moyen extérieur pour pouvoir continuer à utiliser l'automate.

5.2 Simulation de l'automate hybride

Afin de modéliser une stratégie d'achat et de vente à l'aide d'un automate, il a été nécessaire de formuler des hypothèses sur le comportement de notre modèle. Bien que cela n'ait pas été explicitement demandé, il nous a semblé essentiel de réaliser une simulation pour vérifier le bon fonctionnement de notre automate. Sans cette étape, il aurait été difficile de comprendre nos erreurs et d'identifier les points à améliorer pour optimiser cette stratégie.

On obtient ainsi les graphes suivants à l'aide du code de la Figure 6.3 :

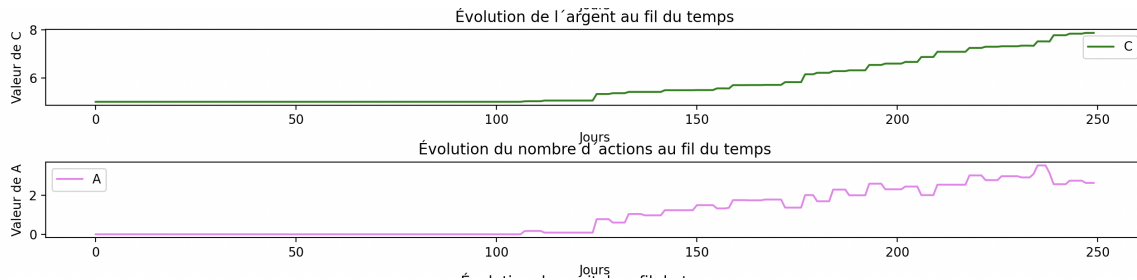


FIGURE 5.2 – Simulation de l'automate sur 250 jours

Pour analyser correctement les performances de l'automate, il suffit d'examiner l'évolution du capital de la banque. Ce dernier est la somme de la quantité d'argent dont dispose l'automate et du produit du nombre d'actions par leur prix unitaire. L'observation de deux courbes croissantes permet de déduire que la stratégie mise en place est adéquate pour le modèle utilisé. Le code utilisé pour simuler les actions de l'automate est disponible en annexe.

Chapitre 6

Conclusion (*1 page max*)

6.1 Conclusion sur les modèles proposés

Pour conclure, notre modèle continu à temps discret permet de modéliser les évolutions des valeurs boursières en utilisant certaines hypothèses relatives au monde de la bourse. Notre modèle possédant 5 paramètres est invariant dans le temps, non-linéaire et possède un point d'équilibre instable. De plus, comme on a pu le voir en 3.1.2 il a été conçu pour être modulable aux différents courts existants. Cependant, il semble reposer sur certaines simplifications excessives pour rendre les équations plus gérables. La réalité des marchés financiers est bien plus complexe et les simplifications pourraient potentiellement négliger des facteurs importants qui influent sur les comportements des agents.

Par la suite, nous avons réalisé un automate hybride qui modélise une stratégie d'achat et vente d'une banque utilisant notre modèle. Notre automate dispose de 2 deux états, un état d'analyse et un état de transaction bancaires en s'appuyant sur notre modèle.

6.2 Perspectives

Comme il est souligné dans la section 3.1.2, pour adapter de manière optimale notre modèle aux dynamiques du marché boursier, il serait préférable d'établir un lien entre la volatilité d'une action et les variations de prix prédites par notre modèle. En identifiant cette corrélation, nous pourrions plus précisément anticiper l'intensité des fluctuations boursières. Cela nous permettrait d'affiner notre stratégie d'achat et de vente, rendant l'automate de trading plus efficace et réactif aux mouvements du marché.

Annexe 1 (si besoin)

Code Python

```
1 import matplotlib.pyplot as plt
2 import numpy as np
3
4 def plot_model(n, P0, S0, proba_changer=0.8):
5     a = 1.2
6     b = 0.5
7     c = 0.4
8     e = 0.55
9     d = 0.1
10
11     vectP = [P0]
12     vectS = [S0]
13     temps = [0]
14     P = P0
15     S = S0
16
17     for i in range(1, n):
18
19         Ptemp = P
20         P = c * S + P - d * P * P * P
21         S = (1 - a * c) * S - a * Ptemp - b * S * S - e * d * Ptemp * Ptemp * Ptemp
22
23         vectP.append(P)
24         vectS.append(S)
25         temps.append(i)
26
27
28     plt.subplot(3, 1, 1)
29     plt.plot(temps, vectP, label='P')
30     plt.title('Evolution de P au fil du temps')
31     plt.xlabel('Jours')
32     plt.ylabel('valeur de P')
33     plt.legend()
34
35     plt.subplot(3, 1, 2)
36     plt.plot(temps, vectS, label='S', color='orange')
37     plt.title('Evolution de S')
38     plt.xlabel('Jours')
39     plt.ylabel('valeur de S')
40     plt.legend()
41
42     plt.subplot(3, 1, 3)
43     plt.plot(temps, vectP, label='P')
44     plt.plot(temps, vectS, label='S', color='orange')
45     plt.title('Superposition de P et S')
46     plt.xlabel('Jours')
47     plt.ylabel('valeur')
48     plt.legend()
49
50     plt.tight_layout()
51
52     # Afficher les graphiques
53     plt.show()
54
55 plot_model(500, -1.585, 0)
```

FIGURE 6.1 – Programme pour obtenir les graphiques du modèle avec ajout de variables aléatoires

```
1 import matplotlib.pyplot as plt
2 import numpy as np
3
4 def plot_model(n, P0, S0, proba_changer=0.8):
5     a = 1.2
6     b = 0.5
7     c = 0.4
8     e = 0.55
9     d = 0.1
10
11     vectP = [P0]
12     vectS = [S0]
13     temps = [0]
14     P = P0
15     S = S0
16
17     for i in range(1, n):
18
19         Ptemp = P
20         P = c * S + P - d * P * P * P
21         S = (1 - a * c) * S - a * Ptemp - b * S * S - e * d * Ptemp * Ptemp * Ptemp
22
23         if np.random.rand() < proba_changer:
24             S = np.random.normal(loc=0, scale=0.40) # Ajouter une fluctuation normale à S
25
26         vectP.append(P)
27         vectS.append(S)
28         temps.append(i)
29
30
31     plt.subplot(3, 1, 1)
32     plt.plot(temps, vectP, label='P')
33     plt.title('Evolution de P au fil du temps avec fluctuations aléatoires')
34     plt.xlabel('Jours')
35     plt.ylabel('valeur de P')
36     plt.legend()
37
38     plt.subplot(3, 1, 2)
39     plt.plot(temps, vectS, label='S', color='orange')
40     plt.title('Evolution de S au fil du temps avec fluctuations aléatoires')
41     plt.xlabel('Jours')
42     plt.ylabel('valeur de S')
43     plt.legend()
44
45     plt.subplot(3, 1, 3)
46     plt.plot(temps, vectP, label='P avec fluctuations aléatoires')
47     plt.plot(temps, vectS, label='S avec fluctuations aléatoires', color='orange')
48     plt.title('Superposition de P et S avec fluctuations aléatoires au fil du temps')
49     plt.xlabel('Jours')
50     plt.ylabel('valeur')
51     plt.legend()
52
53     plt.tight_layout()
54
55     # Afficher les graphiques
56     plt.show()
57
58 plot_model(500, -1.585, 0)
```

FIGURE 6.2 – Programme pour obtenir les graphiques du modèle avec ajout de variables aléatoires

```

1 import matplotlib.pyplot as plt
2 import numpy as np
3
4
5
6
7 def plot_mode(n, P0, S0, capital, constante, prob_change=1):
8     a, b, c, d, e = 1.2, 0.5, 0.4, 0.55, 0.1
9     vectC, vectA, vectP, vectS, vectF = [capital], [0], [P0], [S0], [capital]
10    temps, variations_P, actions = [0], [], 0
11    P, S, deltaP = P0, S0, 0.4
12
13    for i in range(1, n):
14        Ptemp = P
15        P = c * S + P - d * P * P * P
16        S = (1 + e * c) * S - a * Ptemp - b * S * S * S - e * d * Ptemp * Ptemp * Ptemp
17
18        #ajout des non linéarités
19        if np.random.rand() < prob_change:
20            S += np.random.normal(0, 0.1)
21
22        #Maj de Pdelta
23        variation = P - vectP[-1]
24        variations_P.append(abs(variation))
25        if len(variations_P) > 100:
26            variations_P.pop(0)
27        Pdelta = 1.5 * np.mean(variations_P)
28
29        # Logique d'achat
30        if variation < -deltaP and len(variations_P) == 100:
31            quantite_achat = capital * (-deltaP - variation)
32            cout_achat = quantite_achat * P
33            # Achat si le capital est suffisant
34            if cout_achat <= capital:
35                capital -= cout_achat
36                S += quantite_achat/1000
37                actions += quantite_achat
38            else:
39                quantite_achat = capital / P
40                capital = 0
41                S += quantite_achat/1000
42                actions += quantite_achat
43
44        # Logique de vente
45        elif variation > deltaP and len(variations_P) == 100:
46            quantite_vente = capital * (variation - deltaP)
47            # Assurer que la banque ne vend pas plus que ce qu'elle détient
48            if quantite_vente > actions:
49                quantite_vente = actions
50            capital += quantite_vente * P
51            S -= quantite_vente/1000
52            actions -= quantite_vente
53
54        vectP.append(P)
55        vectS.append(S)
56        temps.append(i)
57        vectC.append(capital)
58        vectA.append(actions)
59        vectF.append(capital + actions * vectP[-1])
60
61    plt.subplot(5, 1, 1)
62    plt.plot(temps, vectP, label='P')
63    plt.title('Evolution de P au fil du temps')
64    plt.xlabel('jours')
65    plt.ylabel('Valeur de P')
66    plt.legend()
67
68    plt.subplot(5, 1, 2)
69    plt.plot(temps, vectS, label='S', color='orange')
70    plt.title('Evolution de S au fil du temps')
71    plt.xlabel('jours')
72    plt.ylabel('Valeur de S')
73    plt.legend()
74
75    plt.subplot(5, 1, 3)
76    plt.plot(temps, vectC, label='C', color='green')
77    plt.title('Evolution de l'argent au fil du temps')
78    plt.xlabel('jours')
79    plt.ylabel('Valeur de C')
80    plt.legend()
81
82    plt.subplot(5, 1, 4)
83    plt.plot(temps, vectA, label='A', color='violet')
84    plt.title('Evolution du nombre d'actions au fil du temps')
85    plt.xlabel('jours')
86    plt.ylabel('Valeur de A')
87    plt.legend()
88
89    plt.subplot(5, 1, 5)
90    plt.plot(temps[500:], vectF[500:], label='F', color='pink')
91    plt.title('Evolution du capital au fil du temps')
92    plt.xlabel('jours')
93    plt.ylabel('Valeur de F')
94    plt.legend()
95
96    plt.tight_layout()
97
98    # Afficher les graphiques
99    plt.show()
100
101    plot_mode(250, 0.35, 0.35, 5, 0.4)

```

FIGURE 6.3 – Programme de simulation de l'automate hybride

Bibliographie

- [1] La Théorie de la spéculation, L.Bachelier,1900, *Annales scientifiques de l'E.N.S. 3e serie, tome 17*.
- [2] Optimisation et contrôle stochastique appliques a la finance, H.Pham ,*Springer*.
- [3] The Black–Scholes Model, M Capiński, E Kopp, 2012,,*Cambridge University Press*