



**HAL**  
open science

# Processus de Markov déterministes par morceaux branchants et problème d'arrêt optimal, application à la division cellulaire

Maud Joubaud

► **To cite this version:**

Maud Joubaud. Processus de Markov déterministes par morceaux branchants et problème d'arrêt optimal, application à la division cellulaire. Statistiques [math.ST]. Université Montpellier, 2019. Français. NNT : 2019MONT031 . tel-02378390

**HAL Id: tel-02378390**

**<https://theses.hal.science/tel-02378390>**

Submitted on 25 Nov 2019

**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.

# THÈSE POUR OBTENIR LE GRADE DE DOCTEUR DE L'UNIVERSITÉ DE MONTPELLIER

**En Biostatistiques**

**École doctorale : Information, Structures, Systèmes**

**Unité de recherche : Institut Montpellierain Alexander Grothendieck**

## **Processus de Markov déterministes par morceaux branchants et problème d'arrêt optimal, application à la division cellulaire**

**Présentée par Maud Joubaud**

**Le 25 juin 2019**

**Sous la direction de Benoîte de Saporta  
et Bertrand Cloez**

Devant le jury composé de

Bertrand CLOEZ  
Madalina DEACONU  
Benoîte de SAPORTA  
Nadine HILGERT  
Nathalie KRELL  
Nikolaos LIMNIOS  
Florent MALRIEU

Chargé de recherche, INRA Montpellier  
Chargée de recherche, INRIA Nancy  
Professeure, Université de Montpellier  
Directrice de recherche, INRA Montpellier  
Maître de conférence, Université de Rennes 1  
Professeur, Université de Technologie de Compiègne  
Professeur, Université de Tours

Co-encadrant de thèse  
Rapporteuse  
Directrice de thèse  
Examinatrice  
Examinatrice  
Examineur  
Rapporteur



**UNIVERSITÉ  
DE MONTPELLIER**



# Remerciements

## Résumé

On se propose ici de remercier les différentes personnes ayant eu une importance au cours de toute la thèse, et même avant (et sûrement après). La motivation principale étant qu'on ne remercie jamais assez les personnes qui nous ont aidé, apporté du soutien ou encore de nombreuses choses pour lesquelles une énumération exhaustive serait à la puissance du continu.

*Mots-clé* : Merci, c'est cool, vous gérez les gens.

## 1. Introduction

Dans de nombreuses thèses, on retrouve la présence de remerciements, comme dans [📖] ou encore [📖]. Les structures sous-jacentes sont souvent similaires, c'est pourquoi dans une première partie on tente de construire nos propres remerciements en s'inspirant des structures pré-établies de remerciements, en se basant sur notre propre espace d'état. On présente ensuite une vision plus personnelle de cet espace d'état, en rajoutant une dynamique issue de la théorie des jeux. Les démonstrations des différentes affirmations sont laissées en exercice à la lectrice.

## 2. Structure usuelle des remerciements

Tout d'abord, bien évidemment, je tenais à remercier Benoîte et Bertrand ; il m'est difficile d'expliquer avec des mots clairs et précis toute l'aide, tout le soutien, toute la patience, qu'ils m'ont apporté-e-s. Sans oublier leur rigueur, leur disponibilité, leur bienveillance... Tout cela m'a permis de travailler dans un environnement idéal pendant ces 3 ans.

Un immense merci aussi à Madalina Deaconu et Florent Malrieu pour leur relecture attentive de mon manuscrit, ainsi que leurs conseils précieux et pertinents. Je suis également très honorée de la participation de Nadine Hilgert, Nathalie Krell, Nikolaos Limnios à mon jury de soutenance.

Merci aux membres de MISTEA et de l'IMAG pour m'avoir accueillie en stage à mon arrivée à Montpellier puis pour ma thèse ; l'ambiance était à chaque fois propice à la bonne humeur et l'écoute ce qui m'a permis d'être dans un environnement serein. Trop de noms me viennent en tête pour pouvoir citer chacun-e, c'est pour cela on utilise la propriété suivante :


**Propriété 📖** : *Il n'existe pas de dénombrement possible pour remercier tout le monde en un temps borné.*

Ainsi, je remercie les équipes dans leur globalité, notamment l'équipe administrative (Bernadette, Sophie, Laurence...). Un merci particulier à Alice et Gwladys pour leur accompagnement et leurs ressources dans mes missions d'enseignements à Polytech. En parlant d'enseignement (enseignements → élèves → Math.en.Jeans), merci aussi à toute l'équipe IREM pour leur organisation, leur bonne humeur et tout plein d'autres trucs que j'oublie, ceci se déduisant de la propriété suivante (corollaire de 📖) :

**Propriété 📖** : *L'espace des remerciements n'est pas complet.*

Vient maintenant une partie conséquente de cette section car elle concerne les doctorant-e-s (actuel-le-s et ancien-ne-s). Au vu des propriétés 📖 et 📖, on propose un échantillon de merci à : Dr Abel, co-bureau du début à (quasi) la fin pour son aide, son soutien et nos soirées cuisines trop stylées ; Ivan, co-bureau de 2e génération ; Jérémy (qui restera toujours Schtroumpf grognon, merci Damien pour le surnom), nouveau co-bureau d'en face, pour nos conversations variées plus ou moins sérieuses ; Dr Gautier qui au détour d'une pizza ne savait pas qui j'étais ; Tiffany, bien sûr, comment ne pas remercier Tiffany ! Aussi Pelle de la Calaque propaganda, Alan, Amina, Guillaume, Jocelyn, Julie, Louis (et nos échanges sur le crochet), Meriem, Nicolas, Tom... Merci à vous et à tous les gens que j'oublie, pour m'avoir supportée et pour toute votre aide, votre bonne humeur et la bonne ambiance quasi omniprésente.

Merci aussi à mes ami-e-s IRL, je pense surtout à TeamSyndromePiluleRouge et ConvergENS pour nos réflexions politiques et nos moments de soutien/indignation le tout dans la compréhension.

Merci aussi à mes potes de Rennes : Floflo et ses recettes  et conseils précieux (et pour m'avoir invitée en séminaire), Laulau et son enthousiasme, Cams et notre passion commune pour les pizzas, Caro, Ninon, Maylis, Fred, Tom, Hugo, Jojo, Claire... 4 ans avec une telle promo, c'était trop chouette (surtout pour l'agreg'... ça... le meilleur moment restant les pizz'agreg au rez-de-chaussée de l'IRMAR). En parlant de Rennes, merci également aux différents enseignant·e·s qui ont pu jalonner mon parcours mathématiques ; notamment ceux en probabilités qui m'ont orientée vers ce monde aux paysages variés, et vers ce sujet de thèse. Sans eux, je ne serais probablement pas en train d'écrire ces lignes à Montpellier.



Merci aussi à la famille, surtout mes parents et mon frère et Esma ; je pense par exemple aux déménagements, les "meilleurs" moments où on voit vraiment tout le soutien de sa famille, sauf pour Esma mais c'est un corollaire de :

**Propriété**  : *Un carton est un état absorbant pour un chat de Markov : une fois dedans, il y reste.*


Je termine enfin ce papier par remercier Florent, Gautier et Mathieu pour la joie qu'ils m'apportent chaque jour.




### 3. Dynamique et théorie des jeux

Afin d'apporter un aspect ludique à tout ce texte, voici la concrétisation d'un projet incongru : la réalisation d'un jeu de société illustrant la thèse et les différent·e·s protagonistes y ayant pris part. Là encore, la propriété  s'applique et toutes les personnes ne sont pas représentées dans ce jeu. De plus, afin d'éviter de surcharger ce document de 7 pages couleurs de cartes et de plateau de jeu, on exposera seulement un échantillon du jeu, à savoir les règles du jeu et le plateau. Les différentes cartes à jouer sont disponibles sur ma page personnelle .

### 4. Références

 DIETRICH, G. (2018) *Nouveaux invariants en géométrie CR et de contact*. Université de Montpellier : thèse de doctorat, Mathématiques et modélisation, sous la direction de Herzlich, M.

 LACABANNE, A. (2018) *Catégorification de données  $\mathbb{Z}$ -modulaires et groupes de réflexions complexes*. Université de Montpellier : thèse de doctorat, Mathématiques et modélisation, sous la direction de Bonnafé, C.

 LEMONNIER, F. <https://floriancuisine.wordpress.com>

 JOUBAUD, M. <https://imag.umontpellier.fr/~joubaud/merci-funky.pdf>

**BUT du jeu : accumuler le plus de points à la fin de la partie. Le jeu se termine lorsqu'une personne dépasse la dernière case, n°63 « fin de thèse » (et gagne 10 points). La partie s'arrête alors et les gens comptent leurs points.**

### AVANT DE JOUER

- > Se munir de différents pions/jetons :
    - pour compter les points ou bien les noter quelque part (mini tableau).
    - un jeton pour savoir quel type de dé utiliser (cf carte effet « changement de loi »). Au début de la partie et par défaut, le dé est uniforme.
    - pions pour représenter l'avancement des personnes sur le plateau (les oiseaux).
    - pions pour les cartes oubliées (pions bleus numérotés).
    - pions pour les cartes bibliothèque (pions bicolores à tourner à chaque tour, et pion rose pour les points progressifs).
- Chaque personne pose son pion sur la case 0, **début de la thèse**.

> Les personnes qui jouent ont deux espaces : les cartes piochées qu'elles mettent dans leur main, et un espace devant elles appelé la réserve. Cette réserve permet de poser les cartes avec des effets permanents (cf plus loin), ce sont les cartes actives. La réserve est composée de 7 emplacements (sauf pouvoir spécial).

> Chaque personne pioche 2 cartes PERSONNAGE et une carte EVENEMENT qu'elle garde dans sa main. Ensuite on mélange les cartes PERSONNAGE et les cartes EVENEMENT restantes en une pioche. Puis on désigne qui commence (vote à main levée).

### DEROULEMENT DU JEU

> chaque personne joue à tour de rôle. Un tour se décompose en plusieurs phases :

1) environnement du tour : la personne lance le dé d'environnement (dé 6) qui détermine les effets suivants selon la valeur du dé :

1. ☹ **pause café** : permet de défausser au maximum autant de cartes que de cartes DOCTORANT-E-S/DOCTEUR-E-S posées devant soi ; chaque carte défaussée rapporte 2 points. Inversement, on peut choisir de perdre 2 points pour pouvoir piocher une carte (autant de fois que précédemment)
2. 🏠 **randonnée** : pareil que précédemment avec les cartes FAMILLE
3. π **séminaire** : pareil que précédemment avec les cartes IMAG
4. 📅 **RDV encadrement de thèse** : +3 si Benoîte ou Bertrand et +7 si les 2
5. ★ **soirée débat** : chacun-e pioche une carte chez le-a voisin-e de gauche
6. ✨ **courses** : permet de défausser une carte pour détruire une carte au choix

2) effet de carte : si la personne a des cartes donnant des points à chaque tour (ou tout autre type d'effet régulier), cela se fait au début du tour.

- 3) action : la personne doit faire au plus 2 actions distinctes parmi :
- lancer le dé et avancer son pion sur la case correspondante, en gagnant éventuellement les points indiqués sur la case ;
  - piocher une carte ;
  - poser une carte EVENEMENT ou PERSONNAGE (devant n'importe qui).

Sauf effet de carte, il n'est pas possible de passer son tour (on doit faire au moins une action). Certaines cartes permettent de faire une action supplémentaire ; d'autres ont des effets comptant pour une action, ce qui se rajoute aux choix des 3 actions de base (mais qui ne permettent pas de cumuler 3 actions). On peut avoir au maximum 5 cartes en main, et 7 cartes devant soi dans sa réserve (sauf effet spécial). Si l'on a trop de cartes en main à la fin du tour, on défausse les cartes en trop.

> Pour activer l'effet d'une carte, la personne doit poser la carte : soit devant elle, soit devant une autre personne, soit dans sa réserve. Les cartes ont trois types d'effets : des effets permanents ☹, des effets temporaires ➡ ou des effets immédiats ➡.

- Les cartes à effet immédiat (certaines cartes EVENEMENT), peuvent être jouée sur soi-même ou sur une autre personne : l'effet s'active alors lorsque qu'on pose la carte devant la personne visée, puis la carte est mise dans la défausse (ces cartes n'ont pas besoin d'emplacement disponible pour s'activer).

- Les effets temporaires fonctionnent sur le même principe, sauf que l'effet dure en général un ou plusieurs tours.

- Les cartes à effet permanent (les cartes PERSONNAGE et certaines cartes EVENEMENT) sont actives lorsqu'elles sont posées devant la personne dans sa réserve. Cela rajoute un pouvoir/contrainte à la personne qui dure tant que la carte est dans la réserve. Les cartes de la réserve ne peuvent être ôtées

que par d'autres cartes EVENEMENT ou d'autres pouvoirs. Les cartes permanentes occupent un emplacement (sauf mention contraire).

> Le jeu se termine lorsqu'une personne dépasse la dernière case, n°63 « fin de thèse ». La partie s'arrête alors et les gens comptent leurs points. Pour gagner des points : en tombant sur une case à points, avec une carte événement, avec un effet de personnage. Il est aussi possible de perdre des points, et d'avoir des points négatifs.

> Règles bonus « **publication simultanée** » : si on atteint exactement le même score qu'une autre personne modulo 100, son score retombe à zéro.

### LES CASES DU PLATEAU

> Certaines cases sont munies d'un rond bleu, affichant un nombre de points. Il faut tomber sur la case (et pas juste passer) pour acquérir le nombre de points, sauf pour la dernière case où il suffit de dépasser afin d'obtenir les points. De même pour les carrés rouges, qui ôtent des points.

> On trouve aussi des cases « **changement de sens** ». Lorsqu'une personne tombe sur cette case, le sens de la partie est inversée. Si la partie se déroulait dans le sens des aiguilles d'une montre, elle passera donc dans le sens inverse.

> Il y a également les cases **Vacances** : les vacances c'est bien, mais cela ne fait pas avancer. La personne ne peut ni lancer le dé, ni gagner de point ce tour-ci (mais peut piocher ou poser une carte). Le tour d'après, une de ses actions doit être un déplacement.

> On trouve aussi deux types de zones sur le plateau :

- **période de bibliographie** : permet juste à chaque début de tour dans la zone de piocher une carte (2 zones).

- **période de rédaction** : la rédaction, c'est pas folichon. Chaque lancer de dé est divisé par deux : 1 ou 2 donne 1, 3 ou 4 donne 2, 5 ou 6 donne 3. De plus, si l'on a 3 cartes doctorant-e-s ou plus de posées devant soi, toutes les cartes posées devant soi ont leurs pouvoirs désactivés (jusqu'au début du premier tour hors de la zone).

### LES CARTES

> Précision sur les personnages :

Les cartes donnant des points de façon régulière (« à chaque tour ») se fait toujours au début du tour (avant l'action). Les cartes ayant des effets en combo ne se combinent pas :

ex : Les cartes ABEL et JULIE permettent d'avoir +1 au dé lorsque posées toutes les deux. Et pas un +2 au dé.

- pour les cartes PARENTS, FRANGIN, ESMA : si on a les trois cartes posées, cela fait 2 points de plus par tour.

- pour BENOÎTE et BERTRAND, il faut tomber pile-poil sur la case de fin (et pas juste la dépasser) pour avoir 30 points.

- pour BUREAU 124, les effets peuvent se faire une seule fois, au début du tour. Le principe du 4-21 : on prend 3 dés, et le but est d'obtenir les chiffres 4,2 et 1. On a 3 lancers maximum, et chaque lancer peut concerner un seul dé ou plusieurs. On peut mettre ainsi de côté un ou deux dés, ou tout relancer.

> Précision sur les événements :

- On ne peut pas cumuler les cartes **transport**, ni les cartes **bibliothèque**.

- Si on a une carte **bibliothèque** dans sa réserve : au premier tour commençant en zone « bibliographie », on lance le dé et on ajoute sa valeur en bonus à notre score. Au 2e tour commençant dans la même zone bibliographie, ce bonus devient un malus : on lance le dé et on ôte au score la valeur obtenue. Ainsi, rester longtemps dans la zone n'est pas forcément stratégique. Pour représenter malus/bonus, on dispose le pion bicolore sur sa carte, le faisant tourner à chaque nouveau tour. (rouge=malus, bleu/vert=bonus).

- Pour les cartes OUBLIE-E (qui peuvent mimer une autre carte posée dans n'importe quelle réserve) : on place un pion bleu foncé ayant le numéro correspondant sur la carte mimée, afin de savoir quelle carte on mime.







### REMERCIEMENTS

> Oui. Des remerciements dans les remerciements. Un problème ?

Merci à Gautier et Florent pour leurs idées, à Thibault pour ses retours et idées, à Tiffany dont l'enthousiasme m'a motivée à concrétiser ce qui devait être juste une blague. Merci aux personnes ayant bien voulu tester le jeu : Florent, son frère, Gautier, Alice et Théo. Merci au chat (parce que qu'on ne remercie jamais assez le chat).



TOUR : environnement (dé 6) / effet de carte / 2 actions distinctions parmi *piocher, poser, se déplacer*.

0	DÉBUT DE LA THÈSE						
1	Période de bibliographie						7 -3 DÉMÉNAGEMENT
14	13	12  Changement de sens	11	10	9	8 +2 WORKSHOP	
15 +2 CONFÉRENCE	16	17	18 +2 ÉCOLE D'ÉTÉ AU CIRM	19 -3 DÉMÉNAGEMENT	20 Vacances	21 +3 FIN DE LA 1ère ANNEE	
28	27	26	25	24  Changement de sens	23	22 +2 RENCONTRES DOCTORALES	
29	Période de bibliographie						
	30	31	32	33 +2 COLLOQUE JEUNES PROBABILISTES	34 +2 SÉMINAIRE RENNES	35 +2 CONFÉRENCE INTERNATIONALE	
42 +3 FIN DE LA 2ème ANNEE	41 Vacances	40	39 +2 ÉCOLE D'ÉTÉ AU CIRM	38	37	36  Changement de sens	
43	44 -3 FAUX DÉPART EN VACANCES	45 +2 SÉMINAIRE INRA	46	47	48  Changement de sens	49	
56	55	54 +3 JOURNÉES DOCTORANTES EN PROBA	53	52	51	50 +2 JOURNÉE DES DOCTORANT-E-S	
57	Période de rédaction						
	58	59 Vacances	60  Changement de sens	61	62	63 FIN DE LA THÈSE 	

(1) dé uniforme  $\mathcal{U}(\llbracket 1,6 \rrbracket)$

(2) dé de Poisson  $\mathcal{P}(4)$

(3) dé géométrique  $\mathcal{G}(1/6)$

(4) dé binomial  $\mathcal{B}(8,2/3)$

Jeu de Loi (de proba) de la thèse

# TABLE DES MATIÈRES

<b>Introduction</b>	<b>1</b>
<b>1 Construction de PDMP</b>	<b>7</b>
1.1 Construction historique de Davis	8
1.1.1 Définitions des différents éléments	9
1.1.2 Etapes de construction	12
1.1.3 Propriété de Markov forte du PDMP réel	15
1.2 Construire un PDMP sur un espace de mesure	16
1.2.1 Structure des espace de mesures	17
1.2.2 PDMP à valeur mesure	18
1.3 Exemples de modèles et lien entre différents modèles	22
1.3.1 PDMP sur un espace de mesure, issu de l'exemple jouet	22
1.3.2 PDMP sur $\mathbb{R}^d$ avec $d$ changeant à chaque saut, $(\tilde{X}_t)_t$	24
1.3.3 PDMP augmenté du temps, $(\tilde{X}_t)_t$	25
1.3.4 Cellule marquée et lien avec les PDMP mesure	26
<b>2 Problème d'arrêt optimal</b>	<b>31</b>
2.1 Arrêt optimal et programmation dynamique	33
2.1.1 Opérateur de programmation dynamique	33
2.1.2 Famille de temps d'arrêt $\varepsilon$ -optimaux	35
2.1.3 Caractérisation de la fonction valeur	36
2.2 Comparaison numérique entre la particule marquée et le PDMP mesure	39
2.2.1 Comparaison analytique des opérateurs sur $B_b(E)$ et $B_b(E)$	40
2.2.2 Etude théorique avec l'identité tronquée	42
2.2.3 Comparaison avec la fonction valeur pour le modèle cellule-marquée.	45
2.3 Etude de $g(Z_1)$	46



<b>3 Etude de cas : TCP et approximation</b>	<b>57</b>
3.1 Notations	58
3.2 Description du modèle TCP	61
3.3 Approximation par schéma d'Euler	64
3.4 Erreur forte	69
3.5 Erreur faible : distance de Wasserstein	76
3.5.1 Définition et quelques propriétés	76
3.5.2 Etude de la distance de Wasserstein pour des TCP et leurs schémas	79
3.5.3 Simuler cette erreur	83
<b>Conclusion et perspectives</b>	<b>87</b>
<b>A Codes MATLAB</b>	<b>89</b>
A.1 Pour la construction générale	89
A.2 Pour l'optimisation	90
A.2.1 Différence numérique entre les fonctions valeurs du modèle particule marquée et du modèle population	90
A.2.2 Illustration de $g(Z_1)$	93
A.3 Pour le TCP	98
A.3.1 Construction de processus TCP	98
A.3.2 Approche par schéma d'Euler	100
A.3.3 Estimation erreur forte	105
A.3.4 Estimation erreur faible	106
<b>Bibliographie</b>	<b>109</b>
<b>Notations</b>	<b>112</b>

# INTRODUCTION

Le but de cette thèse est de s'intéresser à une généralisation d'une certaine classe de processus stochastiques en temps continu, introduite par M. H. A. Davis au début des années 80, dans [Dav84]. Son objectif était de construire une théorie globale pour les processus de Markov non-diffusifs. C'est ainsi qu'il définit les processus de Markov déterministes par morceaux, ou plutôt *Piecewise Deterministic Markov Processes*, qu'on abrègera PDMP dans toute la suite de ce manuscrit. Ce sont des processus appartenant à la famille des processus Markoviens. Heuristiquement, un processus de Markov est un processus dont le comportement futur ne dépend que de sa position présente, sans qu'il y ait besoin de connaître son comportement passé. On pourra se référer par exemple à [Sha88] pour la théorie générale sur les processus de Markov. Les PDMP sont également des processus à saut : à des instants aléatoires, le processus subit une discontinuité dans sa trajectoire, prenant une nouvelle valeur déterminée de façon aléatoire elle aussi. Entre ces sauts aléatoires, le processus suit une trajectoire déterministe. Ils font partie de la famille des processus non-diffusifs mais n'en sont pas les seuls représentants ; on peut penser à des processus de Lévy non-continus, comme par exemple une somme d'un mouvement brownien et somme d'un processus de saut pur (voir par exemple [Ber96]).

On peut schématiser la construction d'un PDMP via la figure 1 : le processus part d'un point initial à l'instant  $t = 0$ , case [1]. Ensuite, le processus suit une trajectoire déterministe, case [2], et s'arrête juste avant l'instant du premier saut  $T_1$ , case [3]. Enfin, le processus saute à l'instant de saut sur une nouvelle position, case [4]. Puis l'on recommence ces étapes en prenant comme nouveau point initial la position post-saut.

Les PDMP tel qu'introduits par Davis prennent des valeurs dans des espaces de dimension finie ; on parlera de PDMP fini-dimensionnel. Un tel PDMP peut modéliser la longueur d'une cellule se divisant de façon symétrique au bout d'un temps aléatoire pour donner deux cellules filles identiques. On choisit de suivre une des deux cellules filles, et on recommence. A partir de ce modèle en dimension 1, on peut chercher à modéliser toute une population de cellules avec les PDMP : le but est d'avoir une information pour chaque cellule à chaque instant, et non pas sur une lignée aléatoire. Les PDMP avec le formalisme de Davis permettent de modéliser ce genre de phénomène, mais cela implique

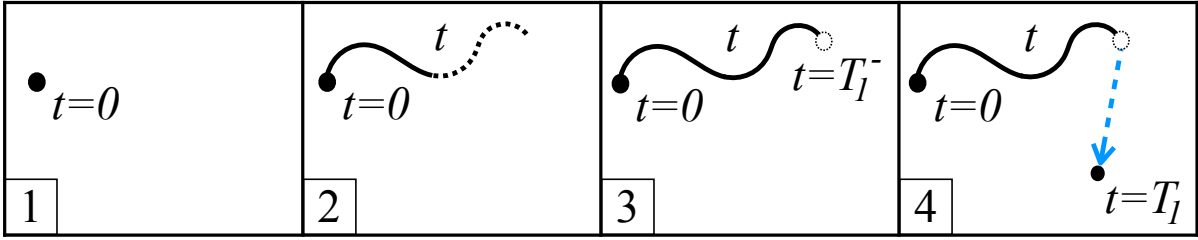


FIGURE 1 – Etapes de construction d'un PDMP, jusqu'au premier saut.

de regarder des unions dénombrables d'espace de dimension finie. Ces espaces ne sont pas pratiques à manipuler, notamment pour des raisons de topologie. Il est possible de se fixer une borne sur la dimension, mais cela contraint notre modèle à avoir un nombre borné de cellules.

Ainsi, on construit un PDMP en dimension infinie. On choisit de se placer sur un espace de mesures discrètes ; une population de cellules est représentée par une somme de masses de Dirac. On dira alors que ces processus sont à valeur mesures, ou encore de processus mesures. De tels PDMP mesures n'ont pas été étudiés, bien qu'on puisse les rapprocher de plusieurs travaux de recherche : par exemple, dans [GT14] des PDMP sont construits sur des espaces de fonctions à valeur dans des espaces Hilbertiens afin de modéliser des systèmes neuronaux. Les PDMP mesures ne rentrent pas dans ce cadre car les espaces de mesures ne correspondent pas à de tels espace d'études. Les dynamiques de population sont également modélisables via des processus de fragmentation, comme dans [Ber06]. Dans ce cadre d'étude, il est étudié des processus à valeurs mesure modélisant une population mais dont le trait physique (la longueur d'une cellule par exemple) est constant entre les sauts. La différence entre ces deux objets mathématiques, processus fragmentation et PDMP mesure, est similaire à la différence entre les processus de saut constants entre les sauts et les PDMP fini dimensionnel. On peut citer également les travaux de [Clo11] sur des processus branchants à valeur mesure (et l'étude de leur comportement limite). Les travaux présentés dans [BDL16] s'intéressent aussi à des processus de fragmentation, pour modéliser des avalanches ; mais ici, contrairement à la division cellulaire, le noyau Markovien n'est pas symétrique.

Cette thèse commence dans un premier temps par expliquer la construction des PDMP finis dimensionnels introduits par Davis, puis notre construction des PDMP à valeur mesure. On y présente la généralisation de la construction de PDMP sur de tels espaces, et également la propriété de Markov forte au théorème 1.2.9. Cela est une partie de notre article soumis [dSCJ18]. On présente plus formellement un modèle de division cellulaire symétrique où l'on s'intéresse à une lignée de cellules choisies aléatoirement parmi les deux filles de la précédente à chaque division ; puis simultanément à la taille de l'ensemble des cellules modélisé par un PDMP à valeur mesure. Le modèle "cellule unique", modélisable grâce à un PDMP de dimension 1, est aussi l'occasion de s'intéresser à des correspondances en loi entre les différents modèles. En effet, il existe des formules, nommées many-to-one, reliant le comportement de la population totale et le comportement d'une cellule marquée. On les présente dans le théorème 1.3.2 avec notre formalisme, c'est-à-dire énoncé avec un PDMP à valeurs mesure qui représente la population totale de cellules. La démonstration quant à elle est issue de [RHK<sup>+</sup>14]. Après avoir défini nos

PDMP à valeur mesure dans la partie 1.2, on s'intéresse à deux problèmes distincts et indépendants.

❖ D'une part, on étudie un problème d'arrêt optimal. L'arrêt optimal est un cas particulier de problème de contrôle stochastique, développé par exemple dans [Dav93, Gug86] pour les PDMP. En considérant un PDMP  $(X_t)_{t \geq 0}$ , le but est de trouver le meilleur temps pour arrêter notre processus, afin de maximiser l'espérance d'une fonctionnelle du processus à l'instant où il est arrêté. Cette fonctionnelle quantifie le gain apporté par notre processus à cet instant, on l'appelle fonction récompense. Le supremum sur les temps d'arrêt de la performance moyenne est appelé fonction valeur. Résoudre un problème d'arrêt optimal revient à déterminer cette fonction valeur.

On peut imaginer la situation suivante : on possède une population de bactéries, modélisée par notre processus mesure  $(X_t)_t$ . Ces bactéries se nourrissent d'une substance toxique présente dans une certaine zone, qu'on cherche à dépolluer. Pour ce faire, on place notre population de bactéries dans la zone polluée, avec un flux continu d'éléments nutritifs pour que la population s'accroisse plus vite et ainsi élimine la substance toxique plus rapidement. Cependant ce flux d'éléments nutritifs a un certain coût : on souhaite que la zone soit dépolluée le plus rapidement possible mais sans avoir un coût trop élevé. On cherche donc à optimiser le temps de dépollution avec un coût moindre : l'arrêt optimal revient à choisir quand arrêter le flux de nutriment (on "ferme" le robinet). On peut citer [Fri14] pour d'autres études de population bactérienne représentée de manière individu-centrée par des mesures aléatoires.

Dans la pratique, on a rarement de manière explicite la fonction valeur, mais il est possible de l'obtenir en itérant une suite d'opérateurs, où intervient la fonction récompense. Il existe de plus des temps d'arrêt explicites pour lesquels, si on stoppe la performance de notre processus, on sera proche de la vraie fonction valeur à  $\varepsilon$  près. On dit alors que de tels temps d'arrêt sont  $\varepsilon$ -optimaux. L'étude des fonctions valeurs sur des PDMP finidimensionnels a été développée par exemple dans [Dav93, Gug86] ou encore [dSDZ15]. Ici, on se propose de l'étendre aux PDMP à valeurs mesures, et de montrer qu'on peut caractériser la fonction valeur par itération d'opérateurs. On exhibe aussi des temps d'arrêt  $\varepsilon$ -optimaux pour notre problème d'arrêt optimal. C'est l'objet de notre théorème 2.1.6, étudié dans notre article [dSCJ18].

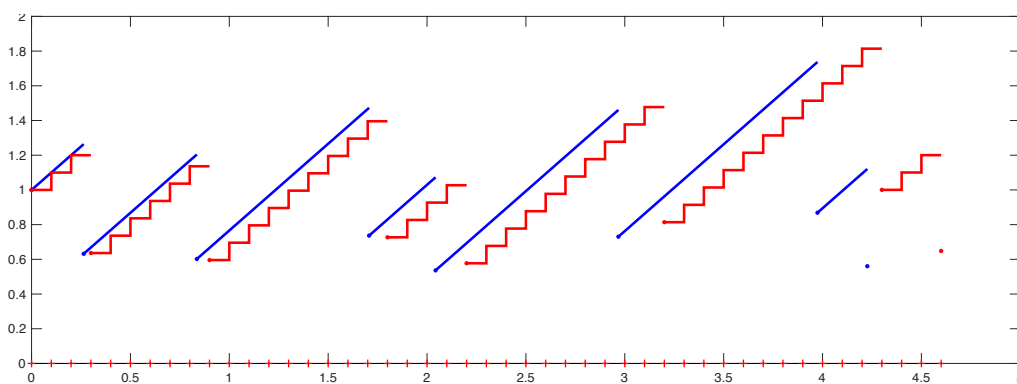
Comme mentionné ci-dessus, les problèmes d'arrêt optimal de PDMP en dimension finie ont déjà été étudiés. De façon plus générale, on pourra regarder [LM84] concernant l'arrêt optimal pour les processus de Markov généraux. En effet, les PDMP font partie des processus markoviens, ce qui rentre dans ce cadre plus général déjà étudié dans la littérature. Cependant, le formalisme des PDMP permet de caractériser chaque PDMP par seulement 3 fonctionnelles, comme présenté dans la sous-section 1.1.1. Leur construction permet à ces objets d'être plus manipulables, notamment pour le problème d'arrêt optimal. On pourrait se demander s'il est possible de résoudre les problèmes d'arrêt optimal sur des PDMP mesure en se ramenant à de la dimension finie, notamment dans le cadre de modèles de population de cellules. En effet, on peut penser aux formules many-to-one, reliant le comportement moyen d'une cellule au comportement moyen d'une population entière. Pour ce faire, on cherche à calculer les différents opérateurs pour les PDMP à valeur mesure dans le cas particulier d'une population de cellules. Plus précisément, le but est de voir s'il était possible de les exprimer en fonction des opérateurs associé au

problème fini-dimensionnel. Les calculs n'amenant pas une relation explicite entre les différents opérateurs, on se base sur des simulations numériques afin d'avoir une preuve empirique de l'équivalence ou non des différents problèmes.

❖ D'autre part, on étudie un PDMP réel particulier modélisant le TCP, *Transmission Control Protocol*, au chapitre 3. Le phénomène est le suivant : on regarde le débit instantané maximal sortant d'un ordinateur connecté à un réseau TCP/IP comme Internet. Plus on avance dans le temps, plus ce débit augmente, de façon déterministe. On le modélise ainsi : l'évolution du débit est linéaire, il augmente d'une unité à chaque pas de temps. Au bout d'un certain temps aléatoire, afin d'éviter la congestion, le débit est divisé par deux. Et le débit évolue à nouveau à partir de cette nouvelle valeur obtenue après le saut. On peut par exemple retrouver cette construction dans [CM16] ; ou encore en tant que cas particulier de [DGR02], article abordant l'étude de comportements asymptotiques de processus markoviens modélisant des phénomènes de congestion informatique. On est dans le cas d'un PDMP en dimension 1. A partir de ce modèle très simple, on cherche à approcher ce processus à l'aide de schéma numérique : en effet, la trajectoire déterministe entre les sauts d'un PDMP peut ne pas être connue explicitement. Le flot, qui la caractérise, n'est parfois connu que via une équation différentielle. Ceci nous porte vers l'étude de schéma numérique, type schéma d'Euler.

La méthode peut se résumer ainsi : on se donne une suite d'instantanés espacés d'une distance  $h$ , formant une grille de discrétisation du temps. On cherche ensuite une valeur d'approximation du flot pris en ces instantanés. En utilisant ce schéma numérique, on construit un autre processus à saut pour approcher le PDMP modélisant le TCP, et on considère un flot approché constant par morceaux sur ces intervalles de longueur  $h$ . Ce processus est construit "en escalier", c'est-à-dire constant par morceaux, sur la grille de discrétisation du temps. On regarde ensuite l'écart entre les deux processus, en étudiant plusieurs types d'erreurs. On peut voir par exemple à la figure 2 l'écart qui augmente entre les deux trajectoires ; l'erreur trajectorielle semble augmenter avec le temps, mais on montre dans le chapitre 3 que cette erreur est majorée. Ces écarts viennent de l'approximation qu'on fait des inter-saut du processus TCP.

On propose des majorations de cette erreur trajectorielle dans le théorème 3.4.1. On s'intéresse également à une erreur "en loi" (dite erreur faible). On présente une approche



**FIGURE 2** – Une trajectoire du TCP en bleu et du processus l'approchant, en rouge.

théorique pour chaque type d'erreur, illustrée par des simulations numériques. La simplicité du modèle permet de mesurer l'écart, essentiellement dû au retard des sauts du processus approchant.

Cette thèse s'organise en trois chapitres principaux. Un premier chapitre présente les PDMP déjà existants ainsi que nos PDMP à valeur mesure. Un second chapitre aborde l'arrêt optimal pour les PDMP à valeurs mesures, fait par programmation dynamique. Enfin, un dernier chapitre sur un PDMP réel fait l'étude de l'approximation du processus par schéma d'Euler et la propagation d'erreurs. Une annexe en fin de manuscrit regroupe les différents codes écrits et utilisés durant la thèse, sous MATLAB, servant aux illustrations numériques (surtout présentes dans les chapitres 2 et 3).

Ce travail a été soutenu par la Région Languedoc-Roussillon et le FEDER, à travers le projet PROMMECE.





# CHAPITRE

## 1

# CONSTRUCTION DE PDMP

### Plan du chapitre

---

<b>1.1 Construction historique de Davis</b> . . . . .	<b>8</b>
1.1.1 Définitions des différents éléments . . . . .	9
1.1.2 Etapes de construction . . . . .	12
1.1.3 Propriété de Markov forte du PDMP réel . . . . .	15
<b>1.2 Construire un PDMP sur un espace de mesure</b> . . . . .	<b>16</b>
1.2.1 Structure des espace de mesures . . . . .	17
1.2.2 PDMP à valeur mesure . . . . .	18
<b>1.3 Exemples de modèles et lien entre différents modèles</b> . . . . .	<b>22</b>
1.3.1 PDMP sur un espace de mesure, issu de l'exemple jouet . . . . .	22
1.3.2 PDMP sur $\mathbb{R}^d$ avec $d$ changeant à chaque saut, $(\check{X}_t)_t$ . . . . .	24
1.3.3 PDMP augmenté du temps, $(\tilde{X}_t)_t$ . . . . .	25
1.3.4 Cellule marquée et lien avec les PDMP mesure . . . . .	26

---

Le but de ce chapitre est d'amener à la construction de PDMP prenant leurs valeurs dans des espaces de mesures, ainsi que de motiver le choix de tels espaces. En effet, à leur formalisation par Davis, les PDMP sont introduits comme des processus à sauts, se comportant entre les sauts de manière déterministe. Il nous faut trois éléments pour décrire ces processus : le comportement déterministe entre les sauts, la loi des temps de sauts, la loi des positions du processus après les sauts. Nous commençons d'ailleurs ce chapitre par exposer la construction des PDMP telle que Davis l'a faite dans [Dav84]. Les PDMP y sont présentés comme des processus hybrides. Ils possèdent deux composantes :



une composante discrète, appelée le mode, et une composante évoluant dans des sous-espaces de  $\mathbb{R}^n$ , appelée la variable d'état. Cette variable d'état représente en général une caractéristique que l'on souhaite étudier, comme la longueur d'une cellule par exemple. A instant où le processus effectue un saut, l'espace où évolue cette variable d'état peut changer, passant par exemple de  $\mathbb{R}^n$  à  $\mathbb{R}^{n+1}$ . La dimension de l'espace où évolue le PDMP n'est pas forcément bornée, des PMDP en dimension infinie ont déjà été développés par exemple dans [BR11b], [GT14] ou encore [RTW12]. Les PDMP y sont à valeurs dans des espaces de Hilbert séparables, afin de modéliser des phénomènes en lien avec des modèles de membranes.

Les PDMP peuvent également modéliser l'évolution de cellules, comme l'exemple fil rouge que nous allons utiliser tout au long de ce chapitre et du chapitre 2. En effet, on peut s'intéresser à l'évolution de la longueur d'une cellule, que représente la variable d'état de notre processus. On suppose que la cellule a une durée de vie aléatoire, et que pendant ce temps sa taille grandit. Le saut du processus représente donc une division cellulaire. On pourrait également regarder la longueur de plusieurs cellules, et à chaque saut (chaque nouvelle division), le nombre de cellules augmente de 1. La dimension de l'espace augmente également de 1. Si on ne veut pas s'imposer une population finie, cela nous oblige à changer d'espace à chaque saut ; d'un point de vue topologique cela ne rend pas ses objets faciles à manipuler.

Dans la deuxième partie de ce chapitre, on introduit une construction des PDMP sur des espaces différents, en vue de la relier à une dynamique de population non forcément bornée. Au lieu de modéliser la dynamique d'un seul individu à l'aide d'un PDMP de dimension finie, le processus prend en compte la dynamique de tous les individus de la population branchante. Une telle population peut être représentée par une mesure discrète, d'où la nécessité de définir des PDMP valorisés par une mesure. La représentation de processus à valeur mesure dans la dynamique des populations est utilisée, par exemple, dans [Ber06, FM04], pour des processus de type fragmentation. Il s'agit d'un cas particulier de PDMP à valeur mesure, sans comportement déterministe entre les sauts et dont la loi des temps de sauts est une loi exponentielle.

Enfin, après avoir construit nos PDMP, on présente quelques modèles issu de la dynamique des population. L'accent est notamment mis sur la définition des éléments caractéristiques du modèle population à partir de celle du modèle individu centré, ceci étant la motivation principale de notre construction sur des espaces de mesures.

## 1.1 Construction historique de Davis

On présente ici la construction des PDMP quasi telle que faite par Davis dans [Dav93], au chapitre 2 (paragraphe 24). En plus de rappeler les PDMP "classiques", cela permet de voir les points problématiques où le passage à espace de mesures impliquera des changements. Cependant, en vue de notre propre construction, on simplifiera les hypothèses de frontières (voir [Dav93], chapitre 2, paragraphe 4, p57). Premier abus de notation : dans toute la suite, on se référera à cette classe de PDMP comme les *PDMP réels* même

si techniquement l'espace d'état n'est pas issu de  $\mathbb{R}$  mais de sous-espaces de  $\mathbb{R}^n$ , comme nous allons le voir. On donne ici des notations générales qui serviront dans la suite de ce chapitre :

**Notation 1.1.1.**

Soit  $W$  un espace topologique.

- ❖ On note  $\mathcal{B}(W)$  sa tribu borélienne,  $\mathcal{B}_b(W)$  le sous-ensemble des boréliens bornés,  $\bar{W}$  sa fermeture et  $\partial W$  sa frontière.
- ❖ Pour  $x$  dans  $W$ , on note  $\delta_x$  la mesure de Dirac en  $x$ .

### 1.1.1 Définitions des différents éléments

On commence par introduire les éléments nécessaires pour définir nos espaces (qui seront le décor de notre étude) :

Soit  $K$  un espace dénombrable et soit  $d : K \rightarrow \mathbb{N}$  une fonction qui envoie  $K$  dans  $\mathbb{N}$  l'ensemble des entiers naturels. Cette fonction  $d$  sert à numéroter les dimensions des espaces où va évoluer notre processus.

Pour chaque  $v \in K$ , soit  $E_v$  un sous espace ouvert de  $\mathbb{R}^{d(v)}$ . On définit ensuite l'espace  $E$  comme suit :

$$E = \bigcup_{v \in K} \{v\} \times E_v = \{(v, \zeta), v \in K, \zeta \in E_v\}. \quad (1.1.1)$$

**Notation 1.1.2.**

Pour  $x = (v, \zeta)$  dans  $E$ , on dira que  $v$  représente le *mode* et  $\zeta$  la *composante d'état*. L'espace  $E$  est un espace hybride. L'espace  $K$  est appelé *espace des modes*. L'espace total  $E$  est l'*espace d'état du processus*.

Par exemple, si on regarde l'évolution de la longueur de cellule dans une population, le mode pourra correspondre au nombre d'individus de la population. La quantité  $\zeta$  représente le "trait", la caractéristique physique qui nous intéresse. Dans l'exemple juste cité, le trait peut être la longueur des cellules. Ceci implique qu'à chaque nouvelle division cellulaire, on incrémente le mode de 1 ; de fait, l'espace des modes n'est (pour le moment) pas supposé borné.

On munit l'espace  $E$  de la tribu  $\mathcal{E}$  engendrée par les sous-espaces  $A \subset E$  ayant la forme suivante :

$$A = \{v\} \times A_v \quad \text{pour } v \in K \text{ et } A_v \text{ un borélien de } E_v. \quad (1.1.2)$$

On souhaite ensuite métriser  $\mathbf{E}$  avec une distance  $\rho$  qui traduirait que deux points sont à distance infinie s'ils n'ont pas le même mode. Pour  $(x_n)_n = (v_n, \zeta_n)_n$  suite de  $\mathbf{E}$  et  $x = (v, \zeta)$  dans  $\mathbf{E}$ , on voudrait l'équivalence suivante :

$$\rho(x_n, x) \xrightarrow[n \rightarrow \infty]{\mathbf{E}} 0 \iff \exists m \in \mathbb{N} \text{ tq } \begin{cases} v_n = v \text{ pour } n \geq m, \\ \zeta_{m+k} \xrightarrow[k \rightarrow \infty]{\mathbf{E}_m} \zeta. \end{cases} \quad (1.1.3)$$

La convergence dans les espaces  $\mathbf{E}_m$  se fait pour la topologie usuelle sur  $\mathbb{R}^{d(m)}$ , i.e. pour la distance euclidienne classique. Pour la distance  $\rho$ , on peut prendre par exemple :

$$\rho(x, x') = \begin{cases} 1 & \text{si } v \neq v' \\ \chi(\zeta - \zeta') & \text{sinon} \end{cases}, \text{ où } \chi(\zeta) = \frac{2}{\pi} \max_{1 \leq j \leq d(v)} \arctan |\zeta_j| < 1. \quad (1.1.4)$$

On introduit la fonction  $\chi$  car ainsi, la distance vaut 1 si les deux points n'ont pas le même mode et sinon la distance est strictement inférieure à 1.

**Proposition 1.1.3.**

L'espace  $(\mathbf{E}, \mathcal{E}, \rho)$  peut être vu comme un sous-espace d'un espace métrique complet séparable, (dit *espace Polonais*).

Cette propriété est détaillée dans [BS78] au chapitre 7; sans entrer dans les détails, cela assure la construction de variables aléatoires particulières (que l'on va voir dans la suite) sur cet espace  $\mathbf{E}$ .

On introduit maintenant les différents éléments caractéristiques qui vont intervenir dans la construction du PDMP. Ces 3 éléments permettent de caractériser entièrement la loi d'un PDMP, ce qui n'est pas le cas des processus de Markov pris au sens général. On peut décrire ces éléments de manière synthétiques, ainsi que leur rôle dans le PDMP et leur interprétation dans la construction du PDMP. Soit  $v \in K$  et  $\zeta \in \mathbf{E}_v$  fixé, on note  $x = (v, \zeta) \in \mathbf{E}$ , correspondant à l'état initial du processus.

**Eléments caractéristiques :**

❖ On se donne un flot continu  $(x, t) \mapsto \phi(x, t) = (v, \phi_v(\zeta, t))$  pour tout  $x = (v, \zeta)$  dans  $\mathbf{E}$  et  $t$  positif. L'ensemble  $(\phi_v)_{v \in K}$  est lui-même une famille de flots avec, pour chaque  $v$  dans  $K$ ,  $\phi_v : \mathbf{E}_v \times \mathbb{R}_+ \rightarrow \mathbf{E}_v$ . Ce flot peut s'interpréter comme les solutions d'une équation différentielle ou famille d'équations différentielles. Pour  $\zeta$  dans  $\mathbf{E}_v$ , la fonction définie sur  $\mathbb{R}_+$  par  $t \mapsto \phi_v(\zeta, t)$  est la solution de cette équation différentielle avec la condition initiale à l'origine  $\zeta$ . Donc  $\phi_v(\zeta, 0) = \zeta$ . Ce flot possède également la propriété de semi-groupe, à savoir :

$$\forall s, t \geq 0, \quad \phi_v(\cdot, t + s) = \phi_v(\phi_v(\cdot, s), t).$$

On note de plus :

$$t^*(x) = \inf\{t > 0, \phi_v(\zeta, t) \in \partial \mathbf{E}_v\} \quad (1.1.5)$$

le temps d'atteinte de la frontière de l'espace  $\mathbf{E}_v$  en partant du point  $x$ , avec la convention classique  $\inf \emptyset = +\infty$ . C'est un temps déterministe, et s'il est infini cela signifie que le processus n'atteint pas la frontière en temps fini.

- ❖ Soit  $l : E \rightarrow \mathbb{R}_+$  une fonction mesurable. Elle vérifie la propriété d'intégrabilité locale le long du flot : pour tout  $x = (v, \zeta) \in E$ , il existe un  $\varepsilon > 0$  tel que

$$\int_0^\varepsilon l \circ \phi(x, s) ds < \infty,$$

où on note  $l \circ \phi(x, s) = l(v, \phi_v(\zeta, s))$  pour tout  $x$  dans  $E$  et  $s > 0$ .

- ❖ Soit  $Q : \bar{E} \times \mathcal{B}(\bar{E}) \rightarrow [0, 1]$  un noyau de transition markovien. Il vérifie donc :
  - pour tout  $A$  dans  $\mathcal{E}$ ,  $Q(\cdot, A)$  est mesurable sur  $(\bar{E}, \mathcal{E})$ ,
  - pour tout  $x$  dans  $\bar{E}$ ,  $Q(x, \cdot)$  est une mesure de probabilité sur  $(E, \mathcal{E})$ .

On impose en plus à  $Q$  de vérifier :

$$\forall x \in E, Q(x, \{x\}) = 0. \quad (1.1.6)$$

Maintenant que nous avons introduit les principaux protagonistes, nous devons introduire les personnages secondaires et leurs différentes relations :

Soit  $x = (v, \zeta)$ . On définit

$$F(x, t) = \mathbb{1}_{\{t < t^*(x)\}} \exp\left(-\int_0^t l \circ \phi(x, s) ds\right). \quad (1.1.7)$$

On reconnaît presque la fonction de survie d'une loi exponentielle dont l'intensité est donnée par la fonction  $l$  composée avec le flot. Le "presque" vient de l'indicatrice : elle traduit le fait qu'on ne peut pas excéder le temps d'atteinte de la frontière par le flot  $\phi$  partant du point  $x$ .

**Notation 1.1.4.**

On note  $\mathcal{E}_x(l \circ \phi)$  la loi dont la fonction de répartition est donnée par  $F(x, \cdot)$ , et on note  $\mathcal{E}(\alpha)$  la loi exponentielle classique de paramètre  $\alpha > 0$ .

Afin de construire des variables aléatoires suivant cette loi, on passe par la pseudo-inverse de cette fonction : pour  $u$  dans  $]0, 1[$  on note

$$\psi_1(x, u) := \inf\{t \geq 0, F(x, t) > u\} = F^{[-1]}(x, u). \quad (1.1.8)$$

Ainsi, pour une variable  $U$  de loi uniforme sur  $[0, 1]$ , la variable  $\psi_1(x, U)$  suit la loi  $\mathcal{E}_x(l \circ \phi)$ .

De même, on veut construire des variables aléatoires dont les lois associées sont données via le noyau de transition, qui n'est pas directement une fonction de survie, se basant sur les boreliens de  $E$ . Afin de clarifier les choses, on note  $\mathcal{L}eb$  la mesure de Lebesgue sur  $\mathbb{R}$ . Il existe une fonction  $\psi_2 : E \times [0, 1] \rightarrow E$  telle que

$$\mathcal{L}eb(\psi_2^{[-1]}(x, A)) = \mathcal{L}eb(\{u \in [0, 1] \mid \psi_2(x, u) \in A\}) = Q(x, A). \quad (1.1.9)$$

L'existence de cette fonction  $\psi_2$  est donnée dans [Dav93]. Ainsi, la variable aléatoire  $\psi_2(x, U)$  suit la loi  $Q(x, \cdot)$ .

Ici, on voit un point qui va être important au moment du passage aux espaces de mesures, ce sont les existences des fonctions  $\psi_1$  et  $\psi_2$  sur ces espaces. En effet, la fonction  $\psi_1$  ne sera pas problématique à généraliser, mais l'existence de  $\psi_2$  telle que justifiée dans [Dav93] (via [BS78]), suppose d'être sur des espaces réels.

Après cette remarque anticipant les problèmes de la section suivante 1.2 on passe à la construction du processus.

### 1.1.2 Etapes de construction

Soit  $(\Omega, \mathcal{A}, \mathbb{P})$  un espace de probabilité. Soit  $(U_1, U_2, \dots)$  une suite de variables aléatoires indépendantes et identiquement distribuées selon la loi  $\mathcal{U}[0, 1[$ . La trajectoire d'un PDMP  $(X_t(\omega)_{t \geq 0})$  prenant valeur dans  $E$  pour  $\omega \in \Omega$  est définie de manière itérative :

- 1 On se donne un point initial  $x_0 = (v_0, \zeta_0)$  dans  $E$  :  $X_0(\omega) = x_0$ .
- 2 On définit  $T_1(\omega) = S_1(\omega) := \psi_1(x_0, U_1(\omega))$  le premier temps de saut du processus. La fonction de survie de  $S_1$  est donnée par  $\mathbb{P}(S_1 > t | X_0 = x_0) = F(x_0, t)$ . Au vu de son expression donnée en (1.1.7), on remarque qu'on a deux types de sauts : les sauts "purement" aléatoires (lorsqu'on est strictement inférieur à  $t^*(x)$ ) et les sauts forcés lorsqu'on touche le bord (voir la figure 1.1.1). La fonction  $l$  intervenant directement comme intensité dans l'exponentielle, elle est appelée *intensité de saut* (ou encore *taux de saut*).
- 3 Sur  $t \in [0, S_1(\omega)[$ , on pose  $X_t(\omega) = \phi_{v_0}(\zeta_0, t)$ . Le flot correspond donc au comportement du processus entre les sauts. D'où le nom de *déterministe par morceaux*.
- 4 A l'instant du premier saut, on pose  $X_{S_1}(\omega) := \psi_2((v_0, \phi_{v_0}(\zeta_0, S_1(\omega))), U_2(\omega))$ . Cette position post-saut est une variable aléatoire, notée  $Z_1 := X_{S_1}$ , qui suit la loi donnée par le noyau markovien  $Q$ . Plus précisément, la loi de  $Z_1$  conditionnellement à  $S_1$  est

$$\mathcal{L}(Z_1 | S_1) = Q(\phi(x_0, S_1), \cdot) = Q(X_{S_1^-}, \cdot) \quad (1.1.10)$$

où  $\mathcal{L}(\cdot | \cdot)$  désigne la loi conditionnelle; de même on notera  $\mathcal{L}(\cdot)$  pour désigner la loi d'une variable aléatoire. Le noyau permet donc de sélectionner les positions post-saut du processus; la propriété (1.1.6) indiquant que le processus ne peut pas sauter sur place.

- 5 On recommence en prenant comme nouveau point initial  $X_{S_1}(\omega) = (v_1, \zeta_1)$ .

L'itération est la suivante : pour  $n \geq 0$

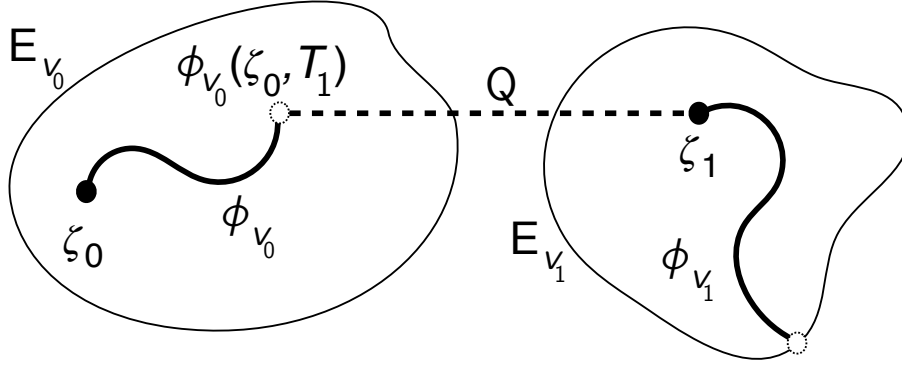
$$S_{n+1}(\omega) = \psi_1(X_{T_n}(\omega), U_{2n+1}(\omega)), \quad T_{n+1}(\omega) = T_n(\omega) + S_{n+1}(\omega),$$

et

$$X_t(\omega) = \phi(X_{T_n}(\omega), t - T_n(\omega)), \quad \text{pour } T_n(\omega) \leq t < T_{n+1}(\omega),$$

$$X_{T_{n+1}}(\omega) = \psi_2((v_{n+1}, \phi_{v_{n+1}}(\zeta_n, S_{n+1}(\omega))), U_{2n+2}(\omega)) = \psi_2(X_{T_{n+1}^-}(\omega), U_{2n+2}(\omega)).$$

On donne un exemple de trajectoire sur la figure 1.1.1, où les deux premiers sous-espaces sont dans  $\mathbb{R}^2$ , partant du point  $(v_0, \zeta_0)$ . Pour  $x$  dans  $E$ , on note  $(X_t^x)_{t \geq 0}$  le PDMP réel dont le point initial est  $x$ . Egalement, on note :



**FIGURE 1.1.1 – schéma d'une trajectoire de point initial  $(v_0, \zeta_0)$ .**  
 Ici, le premier saut est un saut "purement" aléatoire (dans l'espace  $E_{v_0}$  à  $E_{v_1}$ ) et le deuxième saut est provoqué par l'atteinte du bord de  $E_{v_1}$ .

**Notation 1.1.5.**

Les durées inter-sauts successives sont notés  $S_n$  et les temps de sauts notés  $T_n := \sum_{i=1}^n S_i$ , avec la convention  $S_0 = T_0 = 0$ . On note aussi pour  $\omega$  dans  $\Omega$  :

$$N_t(\omega) = \sum_{k \geq 0} \mathbb{1}_{t \geq T_k(\omega)}. \quad (1.1.11)$$

Ces notations des instants de sauts  $T_k$  et des durées inter-sauts  $S_k$  serviront pour tous les modèles présentés dans la suite. Contrairement aux notations pour le processus  $(X_t)_t$  qui changeront à chaque modification majeure d'hypothèses (changement d'espace d'état par exemple), on gardera dans toute la suite  $T_k$  et  $S_k$  ceci afin de ne pas rajouter de nouvelles notations.

Afin d'éviter des problèmes d'explosions, et de pouvoir définir les trajectoires de  $(X_t)_{t \geq 0}$  pour tout  $t$  positif, on fait l'hypothèse suivante :

**Hypothèses 1.1.6.**

Pour tout point initial  $x$  dans  $E$  et pour tout  $t \geq 0$ , on a :

$$\mathbb{E} \left[ \sum_{n=1}^{\infty} \mathbb{1}_{\{T_n < t\}} \mid X_0 = x \right] < \infty.$$

Cette hypothèse permet notamment d'avoir  $T_k$  qui tend vers l'infini presque sûrement. Il est difficile de formuler des conditions générales sous lesquelles cette hypothèse est vérifiée, à cause des liens compliqués entre les éléments caractéristiques  $(\phi, Q, l)$  ; cependant, on se placera toujours dans des cas où cet hypothèse est vérifiée.

La chaîne incluse, i.e. la suite des positions post sauts, est notée  $(Z_n)_n$  avec  $Z_n := X_{T_n}$ . Cette suite augmentée des inter-sauts,  $(Z_n, S_n)_{n \in \mathbb{N}}$  contient toute l'information aléatoire du processus à temps continu.

C'est ici que se clôt la construction de Davis. A partir de cette définition, on peut déduire plusieurs propriétés mais ce qui nous intéresse (et ce qui justifie le  $M$  dans  $PDMP$ ),

c'est la nature Markovienne de ce processus  $(X_t)_t$ . On commence par donner un exemple de PDMP réel, qui servira de fil rouge et de comparaison pour la section suivante 1.2.

**Exemple 1.1.7.** Considérons l'évolution de la taille d'une seule cellule. Au bout d'un certain temps aléatoire, la cellule se divise en deux cellules de taille égale. On suit ensuite l'évolution de la taille d'une des deux cellules filles. Le noyau est donc juste une division déterministe par deux. On se place ici sans changement de mode, on considérera donc que  $x = \zeta$  au lieu de  $x = (v, \zeta)$ .

On se place sur  $\mathbb{R}$  correspondant à l'espace où évolue le trait considéré, ici la taille de la cellule (on pourrait même se restreindre à  $\mathbb{R}_+$ ). On fera donc l'abus  $E = \mathbb{R}$  dans la mesure où l'on oublie le mode. On suppose que la taille des cellules augmente de façon exponentielle, à un taux de croissance  $r > 0$ . Le temps d'atteinte de la frontière est égal à  $+\infty$ ,  $\mathbb{R}$  n'ayant pas de bord atteignable. Enfin, on suppose que l'intensité est une fonction puissance, caractérisée par un exposant  $\alpha$ .

Les éléments caractéristiques sont donc donnés par, pour  $x$  dans  $\mathbb{R}$ ,  $t$  positif et  $A$  dans  $\mathcal{E}$  :

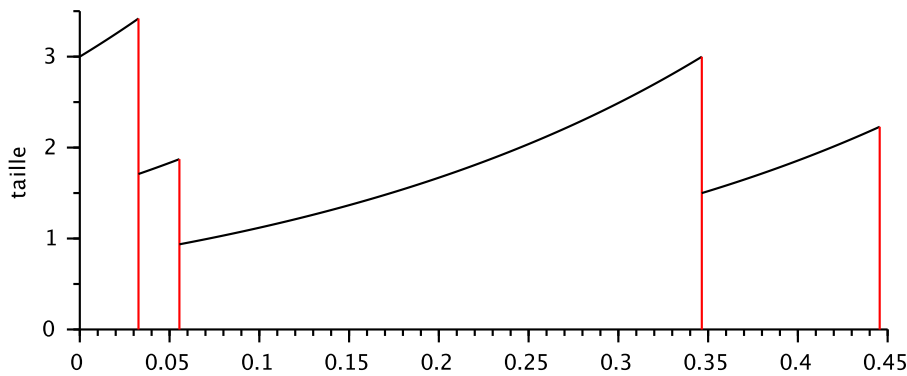
$$\begin{cases} \phi(x, t) = xe^{rt}, \\ Q(x, A) = \delta_{x/2}(A) = \mathbf{1}_A(x/2), \\ l(x) = x^\alpha. \end{cases} \quad (1.1.12)$$

Concernant la simulation, on donne un exemple de trajectoire à la figure 1.1.2. Au vu de la formule (1.1.7), la fonction de survie des temps inter-saut est donnée par :

$$\forall (x, t) \in \mathbb{R} \times \mathbb{R}_+, \quad F(x, t) = \exp\left(-\int_0^t l \circ \phi(x, s) ds\right) = \exp\left(-\frac{x^\alpha}{r\alpha} (e^{rat} - 1)\right)$$

dont la pseudo-inverse est

$$\forall (x, u) \in \mathbb{R} \times ]0, 1[, \quad F^{[-1]}(x, u) = \frac{1}{r\alpha} \ln\left(1 - \frac{r\alpha \ln(u)}{x^\alpha}\right).$$



**FIGURE 1.1.2 – simulation d'une trajectoire d'évolution de la taille d'une cellule.**  
La taille initiale  $x$  vaut 3, le taux de croissance  $r$  vaut 4 et on a aussi  $\alpha$  qui vaut 1. On se réfèrera à l'algorithme A.1.0.1 en annexe.

### 1.1.3 Propriété de Markov forte du PDMP réel

On présente maintenant la propriété markovienne des PDMP, telle qu'exposée dans [Dav93], paragraphe 25.

Soit  $D_E[0, +\infty[$  l'ensemble des fonctions sur  $\mathbb{R}_+$  à valeurs dans  $E$  qui sont continues à droite et admettant des limites à gauche (*càdlàg*). On note  $\overset{c}{X}_t$  l'application coordonnée définie par  $\overset{c}{X}_t(f) = f(t)$  pour  $f$  dans  $D_E[0, +\infty[$ . Soit  $(\mathcal{F}_t^0)_{t \geq 0}$  la filtration naturelle de  $(\overset{c}{X}_t)_t$ , c'est-à-dire  $\mathcal{F}_t^0$  est la tribu engendrée par les  $\{\overset{c}{X}_s, s \leq t\}$  et  $\mathcal{F}^0 = \bigvee_{t \geq 0} \mathcal{F}_t^0$ .

Grâce à l'hypothèse 1.1.6 de non explosion, pour chaque point initial  $x$  dans  $E$  la construction précédente 1.1.2 nous assure l'existence d'une fonction mesurable  $\Upsilon_x$  de  $\Omega$  dans  $D_E[0, +\infty[$  telle que

$$\forall \omega \in \Omega, \quad \overset{c}{X}_t(\Upsilon_x(\omega)) = X_t^x(\omega).$$

On a une correspondance entre les trajectoires possibles (l'ensemble  $D_E[0, +\infty[$ ) et notre construction aléatoire. On note  $\mathbb{P}_x$  la mesure image de  $\mathbb{P}$  par  $\Upsilon_x$ . Cela définit une famille de mesures  $(\mathbb{P}_x)_{x \in E}$  sur  $D_E[0, +\infty[$ . Dans la suite on identifie  $\overset{c}{X}_t$  et  $X_t$ , ce qui nous permet d'écrire :

$$\mathbb{P}_x((X_t)_t \in A) = \mathbb{P}((X_t)_t \in A | X_0 = x).$$

On peut également traduire cet aspect pour une condition initiale non déterministe :

#### Notation 1.1.8.

Pour  $\nu$  une mesure sur  $E$ , on définit la mesure  $\mathbb{P}_\nu$  sur  $(D_E[0, +\infty), \mathcal{F}^0)$  par

$$\forall A \in \mathcal{F}^0, \quad \mathbb{P}_\nu(A) = \int_E \mathbb{P}_x(A) \nu(dx).$$

De plus, pour toute fonction  $f$  mesurable bornée et pour tout  $(x, t)$  dans  $E \times \mathbb{R}_+$ , on note

$$\mathcal{P}_t f(x) = \mathbb{E}_x[f(X_t)] = \int_\Omega f(X_t(\omega)) \mathbb{P}_x(d\omega).$$

On complète la tribu  $\mathcal{F}_t^0$  avec les éléments de  $\mathcal{F}^0$  négligeables pour  $\mathbb{P}_\nu$ , et on note  $\mathcal{F}_t^\nu$  cette complétion. On définit enfin  $\mathcal{F}_t = \bigcap_{\nu} \mathcal{F}_t^\nu$ , l'intersection se faisant sur toutes les  $\nu$  mesures de probabilités sur  $E$ .

Toutes ces notations étant posées, on a le résultat principal de cette partie :

#### Théorème 1.1.9.

Le processus  $(X_t)_{t \geq 0}$  sur  $(D_E[0, +\infty[, (\mathcal{F}_t)_t, (\mathbb{P}_x)_{x \in E})$  est un processus de Markov fort, i.e. vérifiant la propriété suivante : pour tout  $x$  dans  $E$ , pour tout  $T$  ( $\mathcal{F}_t$ )-temps d'arrêt fini et pour toute fonction mesurable bornée  $f$ , on a

$$\forall s > 0, \quad \mathbb{E}_x[f(X_{T+s}) | \mathcal{F}_T] = \mathcal{P}_s f(X_T). \quad (1.1.13)$$



Ce théorème est montré dans [Dav93] d'abord pour la propriété de Markov faible, à savoir montrer l'équation (1.1.13) dans le cas où  $T = t$  est un temps déterministe positif. Puis Davis passe à la propriété forte en utilisant un théorème (présent dans l'annexe du même livre, sur la structure des temps d'arrêt, en A2.3), sur la décomposition de temps d'arrêt.

On passe maintenant à nos PDMP à valeurs mesures, extension des PDMP classiques dans des espaces d'état réels.

## 1.2 Construire un PDMP sur un espace de mesure

Dans cette section, on s'inspire de la construction de Davis mais en passant aux espaces de mesures comme espaces de bases pour notre PDMP. On commence par définir un espace hybride sur lequel la construction de notre processus est possible. Sur cet espace, on introduit les éléments caractéristiques nous donnant la dynamique des trajectoires entre les sauts, les temps de sauts et les positions après saut. Ensuite on montre la propriété de Markov forte et on donne des exemples simples sur des modèles de populations.

Avant toute chose, on donne quelques notations générales.

### Notation 1.2.1.

Soit  $W$  un espace topologique. On note  $B_b(W)$  l'ensemble des fonctions sur  $W$  mesurables bornées à valeurs réelles,  $C_c(W)$  l'ensemble des fonctions à valeurs réelles continues à support compact sur  $W$ .

Et on précise également quelques notations propres aux espaces de mesures.

### Notation 1.2.2.

❖ On note  $\mathfrak{M}$  l'ensemble des mesures localement finies sur  $(\mathbb{R}^m, \mathcal{B}(\mathbb{R}^m))$  et  $\mathfrak{N} = \{\mu \in \mathfrak{M}; \forall B \in B_b(\mathbb{R}^m), \mu(B) \in \mathbb{N}\}$  le sous-ensemble des mesures discrètes. Tout  $\mu$  dans  $\mathfrak{N}$  peut s'écrire comme une somme de mesure de Dirac (possiblement infinie).

❖ Pour toute mesure  $\mu$  dans  $\mathfrak{M}$  et pour toute fonction  $f$  dans  $C_c(\mathbb{R}^m)$ , on pose

$$\mu f := \int_{\mathbb{R}^m} f(x) \mu(dx).$$

❖ On dit qu'une suite  $(\mu_n)_n \subset \mathfrak{M}$  converge vers  $\mu \in \mathfrak{M}$  pour la topologie vague, en notant  $\mu_n \xrightarrow[n \rightarrow \infty]{v} \mu$ , si :

$$\mu_n f \xrightarrow[n \rightarrow \infty]{} \mu f, \quad \forall f \in C_c(\mathbb{R}^m).$$

On munit  $\mathfrak{M}$  de la topologie vague. On remarque alors que  $\mathfrak{N}$  est fermé dans  $\mathfrak{M}$  pour cette topologie (voir par exemple [Kal86]).

### 1.2.1 Structure des espace de mesures

On reprend notre espace des modes  $K$ , ici supposé même fini. De même, pour chaque  $v$  dans  $K$ , soit  $E_v$  un sous-espace ouvert de  $\mathfrak{M}$ , représentant ici aussi l'espace pour la variable d'état associé à  $v$ . L'espace d'état total est noté  $E$  et correspond à :

$$E = \{(v, \mu) \in K \times \mathfrak{M} \mid v \in K, \mu \in E_v\}.$$

On munit cet espace  $E$  de la tribu  $\mathcal{A}$  engendrée par les ensembles  $A$  de la même forme qu'en (1.1.2). Avant de passer à la construction, on donne un exemple simple de l'utilité du choix des PDMP à valeurs mesures, pour modéliser des dynamiques de populations.

**Exemple 1.2.3.** Considérons une population de trois individus à un instant fixé. On note leur taille  $(x_i)_{1 \leq i \leq 3}$ . Alors on peut modéliser cette population dans  $\mathfrak{N}$  via la mesure :

$$\mu = \sum_{i=1}^3 \delta_{x_i} \in \mathfrak{N}.$$

Toute l'information nécessaire sur la taille des individus est contenue dans  $\mu$ . Chaque Dirac dans la somme contient l'information d'un individu, et le nombre de Dirac nous donne le nombre d'individus. Lorsqu'un nouvel individu apparaît (ou disparaît), cela change le nombre de termes dans la somme mais la mesure  $\mu$  reste toujours dans  $\mathfrak{N}$ . Cette façon de représenter les populations est plus simple à manipuler qu'une identification à des vecteurs changeants de taille, notamment pour de la convergence ou des opérateurs itératifs.

On cherche à doter d'une métrique  $\varrho_0$  les espaces topologiques  $\mathfrak{M}$  et  $\mathfrak{N}$ , en vue d'avoir des espaces Polonais, c'est-à-dire des espaces métriques complets et séparables. Cette propriété sera utilisée dans la sous-section 1.2.2. Comme fait dans le livre [Kal86, Appendix], on construit la métrique  $\varrho_0$  comme suit :

Soit  $\mathcal{C}$  une base dénombrable d'ouverts bornés de  $\mathbb{R}^m$ , pour  $m$  dans  $\mathbb{N}$ . Supposons (sans perdre de généralités) que  $\mathcal{C}$  est stable par union finie. Pour tout  $C$  dans  $\mathcal{C}$ , il existe une suite  $(C_n)_n$  dans  $\mathcal{B}_b(\mathbb{R}^m)$  et une suite croissante de fonctions  $(f_{C,n})_n$  dans  $C_c(\mathbb{R}^m)$  telles que :

$$f_{C,n} \xrightarrow{n \rightarrow \infty} \mathbb{1}_C \quad \text{et} \quad \mathbb{1}_{C_n} \leq f_{C,n} \leq \mathbb{1}_C.$$

Comme  $\mathcal{C}$  est dénombrable, l'ensemble  $\{f_{C,n} \mid C \in \mathcal{C}, n \in \mathbb{N}\}$  est dénombrable. On numérote  $f_1, f_2, \dots$  toutes ces fonctions. Et alors une mesure  $\mu$  quelconque est complètement déterminée par l'ensemble  $\{\mu f_k, k \in \mathbb{N}\}$ .

Pour toutes mesures  $\mu$  et  $\mu'$  dans  $\mathfrak{M}$ , on définit la distance  $\varrho_0$  par :

$$\varrho_0(\mu, \mu') := \sum_{k \geq 1} \frac{1}{2^k} (1 - \exp(-|\mu f_k - \mu' f_k|)).$$

On a donc la propriété suivante de structure des espaces pour la variable d'état :

#### Proposition 1.2.4.

Les espaces  $(\mathfrak{M}, \varrho_0)$  et  $(\mathfrak{N}, \varrho_0)$  sont des espaces Polonais : des espaces métriques complets séparables pour la topologie induite par la distance  $\varrho_0$ .

La démonstration se trouve dans le livre [Kal86], toujours dans l'annexe.

Maintenant, on peut s'intéresser à la construction à proprement parler de notre PDMP à valeurs mesure. La démarche reprend le même schéma que dans la section précédente, avec les justification pour que cela fonctionne.

## 1.2.2 PDMP à valeur mesure

On peut maintenant munir l'espace  $E$  d'une métrique, construite de façon similaire à (1.1.4), et qui traduira le fait que deux points qui ne sont pas dans le même sous-espace (donc pas avec le même mode) sont à distance inatteignable.

Soit  $\mathfrak{x} = (v, \mu)$  et  $\mathfrak{x}' = (v', \mu')$  dans  $E$ . On pose :

$$\varrho(\mathfrak{x}, \mathfrak{x}') := \begin{cases} \frac{2}{\pi} \arctan(\varrho_0(\mu, \mu')) & \text{si } v = v', \\ 1 & \text{sinon.} \end{cases} \quad (1.2.1)$$

On retrouve la même équivalence de convergence qu'en (1.1.3), à savoir qu'une suite dans  $E$  converge si et seulement si la suite des modes est constante à partir d'un certain rang, puis la suite des variables d'état converge dans le sous-espace associé pour la métrique  $\varrho_0$ .

On présente maintenant une propriété de l'espace

$$\overline{E} := \{\mathfrak{x} = (v, \mu) \in K \times \mathfrak{M} \mid v \in K, \mu \in \overline{E_v}\}, \quad (1.2.2)$$

où  $\overline{E_v}$  est la fermeture de  $E_v$  pour la distance  $\varrho_0$ . Cet espace  $\overline{E}$  est lui-même la fermeture de  $E$  pour la métrique  $\varrho$ .

### Lemme 1.2.5.

L'espace métrique  $(\overline{E}, \varrho)$  est un espace Polonais.

Cette propriété est nécessaire pour la construction explicite des positions post-saut.

**Preuve.** On propose la démonstration suivante : on doit prouver que  $\overline{E}$  est un espace métrique complet séparable. On commence par montrer que c'est un espace séparable.

Les espaces  $E_v$  sont des sous espaces de  $\mathfrak{M}$ , lui même Polonais pour  $\varrho_0$ . Ce sont donc des espaces à base dénombrable. Comme  $K$  est lui même dénombrable, l'espace métrique  $\overline{E}$  est à base dénombrable (donc séparable) au vu de son écriture en (1.2.2). Reste à voir la complétude.

Pour tout  $v$  dans  $K$ , chaque espace  $(\overline{E_v}, \varrho_0)$  est un espace complet en tant que fermé de l'espace complet  $\mathfrak{M}$ . Soit  $(\mathfrak{x}_n)_n = (v_n, \mu_n)_n$  une suite de Cauchy dans  $(\overline{E}, \varrho)$  et montrons qu'elle est convergente. Elle vérifie :

$$\forall \varepsilon > 0 \quad \exists N \in \mathbb{N}, \quad \forall p, q > N \quad \varrho(\mathfrak{x}_p, \mathfrak{x}_q) < \varepsilon.$$

On prend un  $\varepsilon_0 < 1$ , et on note  $N_0$  le rang associé. On a donc la distance  $\varrho(\mathfrak{x}_p, \mathfrak{x}_q)$  strictement plus petite que 1 et ainsi que  $v_p = v_q$  pour tout  $p, q > N_0$ , d'après la définition de  $\varrho$  en (1.2.1). La suite  $(v_n)$  est donc constante à partir du rang  $N_0$ ; on note  $v$  sa limite.

La suite  $(\mu_n)_{n>N_0}$  évolue donc dans l'espace  $(E_v, \varrho_0)$ . Or, cette suite est aussi de Cauchy pour cet espace : pour n'importe quel  $\varepsilon < \varepsilon_0$ , il existe un rang  $N > N_0$  vérifiant la propriété de Cauchy. Ce rang  $N$  correspond au rang  $N_{\tan(\varepsilon\pi/2)}$ , car on a bien pour tout  $p, q > N_{\tan(\varepsilon\pi/2)}$ ,

$$\varrho(\mathfrak{x}_p, \mathfrak{x}_q) = \frac{2}{\pi} \arctan(\varrho_0(\mu_p, \mu_q)) < \tan(\varepsilon\pi/2) \implies \varrho_0(\mu_p, \mu_q) < \varepsilon.$$

Or,  $(\overline{E}_v, \varrho_0)$  est complet, donc cette suite converge vers un élément  $\mu$ . La suite  $(\mathfrak{x}_n)_n$  est donc convergente dans  $(\overline{E}, \varrho)$ , car la suite  $(v_n)_n$  est constante à partir du rang  $N_0$  et que la suite  $(\mu_n)_n$  converge dans  $(\overline{E}_v, \varrho_0)$  à partir de ce rang  $N_0$ . On a donc montré la convergence de toute suite de Cauchy et ainsi l'espace est complet.  $\square$

Cette propriété étant montrée, on définit de la même façon que précédemment les éléments caractéristiques, avec les mêmes types de propriétés que les éléments caractéristiques des PDMP réels :

- ❖ une famille de flots continus  $(\Phi_v)_{v \in K}$  avec  $\Phi_v : E_v \times \mathbb{R}_+ \rightarrow E_v$  avec la propriété de semi-groupe, ainsi que le temps d'atteinte  $t^*(\mathfrak{x}) = \inf\{t > 0, \Phi_v(\mu, t) \in \partial E_v\}$  ;
- ❖ un taux de saut mesurable  $\lambda : E \rightarrow \mathbb{R}_+$  localement intégrable le long du flot ;
- ❖ un noyau de transition markovien  $Q : \overline{E} \times \mathcal{B}(\overline{E}) \rightarrow [0, 1]$  , sans saut sur place.

La trajectoire d'un PDMP  $(X_t(\omega)_{t \leq 0})$  prenant valeurs dans  $E$  pour  $\omega \in \Omega$  est définie de manière itérative :

- 1 On se donne un point initial  $\mathfrak{x}_0 = (v_0, \mu_0)$  dans  $E : X_0(\omega) = \mathfrak{x}_0$ .
- 2 On veut définir  $T_1(\omega) = S_1(\omega)$  le premier temps de saut du processus. La fonction de survie de  $S_1$  est égale à

$$F(\mathfrak{x}_0, t) := \mathbb{P}(S_1 > t | X_0 = \mathfrak{x}_0) = \mathbb{1}_{\{t < t^*(\mathfrak{x})\}} \exp\left(-\int_0^t \lambda(v, \Phi_v(\mu, s)) ds\right).$$

Comme précédemment, on peut considérer la pseudo-inverse  $F^{[-1]}$ , noté  $\Psi_1$ , et poser  $S_1(\omega) = \Psi_1(\mu_0, U_1(\omega))$ , qui aura la bonne fonction de survie.

- 3 Sur  $t \in [0, S_1(\omega)[$ , on pose  $X_t(\omega) = \Phi_{v_0}(\mu_0, t)$ .
- 4 A l'instant du premier saut, on voudrait que  $X_{S_1}(\omega)$  suive la loi donnée par  $Q(X_{S_1}, \cdot)$ . Or, la fonction  $\Psi_2$  ne s'obtient pas directement comme la fonction  $\Psi_1$  (qui correspond juste à une pseudo-inverse). On a besoin pour cela du lemme suivant, issu de [Kol11, 1.1(p6)] :

**Lemme 1.2.6.**

Soit  $\{\nu(w, dy)\}_{x \in X}$  une famille de probabilité sur un espace  $\mathcal{Y}$  Polonais, dépendant mesurablement d'un paramètre  $w \in \mathcal{W}$  où  $\mathcal{W}$  est un espace polonais. Alors il existe  $f : \mathcal{W} \times [0, 1] \rightarrow \mathcal{Y}$  mesurable telle que :

$$\theta \sim \mathcal{U}[0, 1] \implies f(w, \theta) \sim \nu(w, \cdot).$$

Dans notre cas, les espaces  $\mathscr{W}$  et  $\mathscr{Y}$  correspondent à  $\overline{E}$  qui est bien polonais et  $\nu(w, dy) = Q(w, dy)$ . On pose  $\Psi_2 = f$  et

$X_{S_1}(\omega) := \Psi_2((v_0, \Phi_{v_0}(\mu_0, S_1(\omega))), U_2(\omega))$ . La position post saut est une variable aléatoire, notée  $Z_1$ , qui suit la loi donnée par le noyau markovien  $Q$ . Plus précisément, la loi de  $Z_1$  conditionnellement à  $S_1$  est (comme en (1.1.10)) :

$$\mathcal{L}(Z_1|T_1) = Q(\Phi(\mathfrak{x}_0, S_1), \cdot) = Q(X_{S_1^-}, \cdot) \quad (1.2.3)$$

**5** On recommence en prenant comme nouveau point initial  $X_{S_1}(\omega) = (v_1, \mu_1)$ .

L'itération est la suivante : pour  $n \geq 0$

$$S_{n+1}(\omega) = \Psi_1(X_{T_n}(\omega), U_{2n+1}(\omega)), \quad T_{n+1}(\omega) = T_n(\omega) + S_{n+1}(\omega),$$

et

$$\begin{aligned} X_t(\omega) &= \phi(X_{T_n}(\omega), t - T_n(\omega)) \quad \text{pour } T_n(\omega) \leq t < T_{n+1}(\omega), \\ X_{T_{n+1}}(\omega) &= \Psi_2(X_{T_{n+1}^-}(\omega), U_{2n+2}(\omega)). \end{aligned}$$

On reprend également l'hypothèse 1.1.6 sur la non explosion locale du processus de comptage associé. Enfin, de même qu'à la section 1.1.3, on construit une filtration  $(\mathcal{F}_t)_t$ , une famille de probabilités  $(\mathbb{P}_{\mathfrak{x}})_{\mathfrak{x} \in E}$  avec lesquelles on peut considérer notre PDMP-mesure comme un processus sur  $(D_E[0, +\infty[, (\mathcal{F}_t)_t, (\mathbb{P}_{\mathfrak{x}})_{\mathfrak{x} \in E})$ .

En calquant les notations de la partie précédente, on obtient également la propriété de Markov pour notre processus. La démonstration est similaire à celle de Davis, le point bloquant étant la structure des temps d'arrêt, qu'on adapte ici :

**Théorème 1.2.7.**

Une variable aléatoire positive  $\tau$  est un  $(\mathcal{F}_t)_{t \geq 0}$ -temps d'arrêt si et seulement si il existe une suite  $(R_n)_{n \in \mathbb{N}}$  de variables aléatoires  $(\mathcal{F}_{T_n})_{n \in \mathbb{N}}$ -adaptées telles que :

$$\tau = \sum_{n=1}^{\infty} R_{n-1} \wedge S_n.$$

**Preuve.** La démonstration suit le même enchaînement que dans [dSDZ15, Section 1.7]. Elle est basée sur le théorème A2.3 dans [Dav93] qui s'applique pour tout processus constant par morceau et continu à droite, à valeurs dans un espace de Borel<sup>1</sup> et la bijection entre notre processus  $(X_t)_t$  et le processus  $(\eta_t)_t$  constant par morceau continu à droite à valeurs dans  $E \times \mathbb{N}$  défini par

$$\eta_t = (X_0, 0), \quad t < T_1, \quad \eta_t = (X_{T_n}, n), \quad T_n \leq t < T_{n+1}.$$

□

1. un espace de Borel est un espace mesurable séparé à base dénombrable, ce qui est équivalent à un espace mesurable isomorphe à un espace métrique séparable muni d'une tribu borélienne.

Ici, la filtration  $(\mathcal{F}_t)_t$  est construite à partir de l'espace des fonctions *càdlàg* sur  $E$ , et non pas sur  $E$  comme dans la partie 1.1.3. De même qu'à la notation 1.1.8, on introduit la quantité suivante :

**Notation 1.2.8.**

Pour toute fonction  $f$  mesurable bornée et pour tout  $(\mathbf{x}, t)$  dans  $E \times \mathbb{R}_+$ , on note

$$P_t f(\mathbf{x}) = \mathbb{E}_{\mathbf{x}}[f(X_t)] = \int_{\Omega} f(X_t(\omega)) \mathbb{P}_{\mathbf{x}}(d\omega).$$

Une fois ce théorème 1.2.7 en main, on peut énoncer la propriété de Markov :

**Théorème 1.2.9.**

Le processus  $(X_t)_{t \geq 0}$  sur  $(D_E[0, +\infty[, (\mathcal{F}_t)_t, (\mathbb{P}_{\mathbf{x}})_{\mathbf{x} \in E})$  est un processus de Markov fort, i.e. vérifiant la propriété suivante : pour tout  $\mathbf{x}$  dans  $E$ , pour tout  $T$   $(\mathcal{F}_t)$ -temps d'arrêt et pour toute fonction mesurable bornée  $f$ , on a

$$\forall s > 0, \quad \mathbb{E}_{\mathbf{x}}[f(X_{T+s}) \mathbb{1}_{T < \infty} | \mathcal{F}_T] = P_s f(X_T) \mathbb{1}_{T < \infty}. \quad (1.2.4)$$

**Preuve.** La démonstration se fait de la même façon que [Dav93, Section 25], grâce au théorème 1.2.7.

En effet, on commence par montrer la propriété de Markov faible, i.e. pour  $T = t$  un temps déterministe. Cette propriété vient de la loi des durées inter-saut presque exponentielle. Par calcul, on trouve :

$$\mathbb{P}_{\mathbf{x}}(\inf\{u > t ; X_u \neq X_{u-}\} > t + s | \mathcal{F}_t) = \mathbb{P}_{X_t}(T_1 > s). \quad (1.2.5)$$

Ceci signifie que la loi du prochain saut après  $t$  est la même que celle du premier saut d'un PDMP partant de  $X_t$ . De plus, la loi du processus redémarré à  $X_t$  ne dépendant que de variables aléatoires uniformes indépendantes de tout le reste, d'où la propriété de Markov faible.

Pour la propriété de Markov forte, on écrit notre temps d'arrêt avec la décomposition du théorème 1.2.7. Puis on peut se ramener à la formule (1.2.5) et par les mêmes types d'arguments on montre la propriété de Markov forte.

□

Après avoir vu les PDMP à valeurs mesures, la possibilité de construction et la propriété de Markov, on passe dans la partie suivante à des exemples de modèles de PDMP. Et plus précisément comme relier des modèles PDMP réels, basés sur le fil rouge de la croissance de cellules, à des modèles de population totale, via les PDMP mesure.

## 1.3 Exemples de modèles et lien entre différents modèles

### 1.3.1 PDMP sur un espace de mesure, issu de l'exemple jouet

Dans les cas suivants, on se place sur des espaces sans frontières, donc un temps d'atteinte  $t^*$  infini.

Pour expliciter les caractéristiques locales de ce PDMP mesure (pour la population totale des cellules), on regarde une population initiale composée de  $n$  cellules. La cellule  $i$  a une taille initiale  $x_i$ . Elles ont chacune un temps de vie exponentiel, dépendant de leur état initial comme décrit dans la notation 1.1.4, et elles sont indépendantes. Un évènement arrive quand la cellule "la plus rapide" se divise; on regarde donc la loi minimum entre plusieurs exponentielles.

#### Lemme 1.3.1.

Soit  $\mathcal{T}_1, \dots, \mathcal{T}_n$  des variables aléatoires indépendantes telles que

$$\forall 1 \leq i \leq n, \quad \mathcal{T}_i \sim \mathcal{E}_{x_i}(l \circ \phi).$$

Alors la variable  $M_n := \min\{\mathcal{T}_1, \dots, \mathcal{T}_n\}$  a pour fonction de survie

$$\mathbb{P}(M_n > t) = \exp\left(-\int_0^t \sum_{i=1}^n l \circ \phi(x_i, s) ds\right). \quad (1.3.1)$$

De plus, on note  $A_n$  la variable aléatoire discrète représentant le numéro du minimum des  $(\mathcal{T}_i)_{1 \leq i \leq n}$ , c'est-à-dire  $A_n := \operatorname{argmin}\{\mathcal{T}_1, \dots, \mathcal{T}_n\}$ . Conditionnellement à la valeur du temps minimum, sa loi est donnée pour tout  $j$  dans  $\{1, \dots, n\}$  et tout  $t > 0$  par :

$$\mathbb{P}(A_n = j | M_n = t) = \frac{l \circ \phi(x_j, t)}{\sum_{i=1}^n l \circ \phi(x_i, t)}. \quad (1.3.2)$$

**Preuve.** La fonction de survie (1.3.1) se calcule par indépendance des  $\mathcal{T}_i$  et par propriété de la fonction exponentielle (transformant les sommes en produit); on le démontre de la même façon que pour le calcul de la loi d'un minimum de loi exponentielle.

On voit directement que le support de la variable  $A_n$  est  $\{1, \dots, n\}$ . La loi conditionnelle  $\mathbb{P}_{A_n}(\cdot | M_n = t)$ , pour  $t > 0$ , peut donc s'écrire :

$$\mathbb{P}_{A_n}(\cdot | M_n = t) = \sum_{j=1}^n \mathbb{P}(A_n = j | M_n = t) \delta_j(\cdot).$$

On cherche donc une sorte de "densité" conditionnelle. Pour tout  $1 \leq j \leq n$  et pour tout  $t > 0$ , on peut écrire :

$$\mathbb{P}(A_n = j | M_n = t) = \frac{f_{A_n, M_n}(j, t)}{f_{M_n}(t)},$$

où  $f_{M_n}$  est la densité du minimum des  $\mathcal{T}_i$ , obtenue grâce à la formule (1.3.1), par dérivation :

$$f_{M_n}(t) = \left( \sum_{i=1}^n l \circ \phi(x_i, t) \right) \times \exp \left( - \int_0^t \sum_{i=1}^n l \circ \phi(x_i, s) ds \right).$$

Avant de poursuivre, on remarque l'égalité des évènements suivants :

$$\{A_n = j\} = \left\{ \operatorname{argmin}_{1 \leq i \leq n} \mathcal{T}_i = j \right\} = \left\{ \mathcal{T}_j = \min_{1 \leq i \leq n} \mathcal{T}_i \right\} = \{ \mathcal{T}_j = M_n \}. \quad (1.3.3)$$

Pour calculer la "densité" jointe  $f_{A_n, M_n}$ , on commence par calculer la quantité suivante, pour tout  $t > 0$  et  $1 \leq j \leq n$  :

$$\begin{aligned} \mathbb{P}(A_n = j, M_n > t) &= \mathbb{P}(\forall i \mathcal{T}_i \geq \mathcal{T}_j, \mathcal{T}_j > t) \quad \text{par (1.3.3)} \\ &= \mathbb{E} \left[ \prod_{i=1, i \neq j}^n \mathbb{1}_{\mathcal{T}_i \geq \mathcal{T}_j} \times \mathbb{1}_{\mathcal{T}_j > t} \right] \\ &= \mathbb{E} \left[ \mathbb{E} \left[ \prod_{i=1, i \neq j}^n \mathbb{1}_{\mathcal{T}_i \geq \mathcal{T}_j} \middle| \mathcal{T}_j \right] \mathbb{1}_{\mathcal{T}_j > t} \right] \\ &= \mathbb{E} \left[ \prod_{i=1, i \neq j}^n \exp \left( - \int_0^{\mathcal{T}_j} l \circ \phi(x_i, s) ds \right) \mathbb{1}_{\mathcal{T}_j > t} \right] \\ &= \mathbb{E} \left[ \exp \left( - \int_0^{\mathcal{T}_j} \sum_{i=1, i \neq j}^n l \circ \phi(x_i, s) ds \right) \mathbb{1}_{\mathcal{T}_j > t} \right] \\ &= \int_t^\infty \exp \left( - \int_0^u \sum_{i=1, i \neq j}^n l \circ \phi(x_i, s) ds \right) \times f_{\mathcal{T}_j}(u) du, \end{aligned}$$

où  $f_{\mathcal{T}_j}$  est la densité de  $\mathcal{T}_j$ , donnée par  $u \mapsto (l \circ \phi(x_j, u)) \times \exp \left( - \int_0^u l \circ \phi(x_j, s) ds \right)$ . On a donc :

$$\mathbb{P}(A_n = j, M_n > t) = \int_t^\infty l \circ \phi(x_j, u) \times \exp \left( - \int_0^u \sum_{i=1}^n l \circ \phi(x_i, s) ds \right) du.$$

On en déduit ainsi que  $f_{A_n, M_n}(j, u) = l \circ \phi(x_j, u) \times \exp \left( - \int_0^u \sum_{i=1}^n l \circ \phi(x_i, s) ds \right)$  pour tout  $j, u$ . Et donc, en simplifiant la fraction  $\mathbb{P}(A_n = j | M_n = t)$  par les termes en exponentielle, on trouve l'équation (1.3.2).  $\square$

De cette propriété, on en déduit la forme des éléments caractéristiques, afin d'avoir une cohérence entre le modèle cellule unique, et le modèle population.

On suppose ici encore, comme dans l'exemple 1.1.7, que le mode ne change pas (on ne considère pas de mode). On fera donc l'abus de notation  $\mathfrak{x} = \mu \in \mathfrak{N}$ . On note  $\mu = \sum_{i=1}^n \delta_{x_i}$  et  $\nu = \sum_{i=1}^m \delta_{y_i}$  deux mesures dans  $\mathfrak{N}$ . On pose :

$$\begin{aligned} \lambda &: \mathfrak{N} \rightarrow \mathbb{R} \\ \mu &\mapsto \sum_{i=1}^n l(x_i) \end{aligned} \quad ,$$



$$\begin{aligned} \Phi &: \mathfrak{N} \times \mathbb{R}_+ \rightarrow \mathfrak{N} \\ (\mu, t) &\mapsto \sum_{i=1}^n \delta_{\phi(x_i, t)} \end{aligned}$$

$$Q(\nu, A) = \sum_{j=1}^m \frac{l(y_j)}{\sum_{i=1}^m l(y_i)} \mathbb{1}_A(\nu_{(j)}) \quad \text{avec } \nu_{(j)} = \sum_{i=1, i \neq j}^m \delta_{y_i} + 2\delta_{y_j/2}.$$

Le choix de  $\lambda$  vient directement du lemme 1.3.1, où la loi du premier temps de saut du PDMP mesure correspond à la division de la première cellule. Le flot  $\Phi$  exprime le fait qu'on applique le même flot à chaque cellule de manière indépendante; chaque quantité associée à un Dirac correspond à l'information d'une cellule. Enfin, le noyau  $Q$  nous dit qu'on passe d'une population représentée par la mesure  $\nu$  à  $\nu_{(j)}$  avec probabilité  $l(y_j)/(\sum l(y_i))$ . Cela correspond au fait que c'est la  $j^e$  cellule qui s'est divisée (avant les autres).

On note  $T_1 \sim \text{Exp}_\mu(\lambda \circ \Phi)$  pour signifier :

$$\begin{aligned} \mathbb{P}_\mu(T_1 > t) &= \mathbb{1}_{t < t^*(\mu)} \exp\left(-\int_0^t \lambda \circ \Phi(\mu, s) ds\right) \\ &= \mathbb{1}_{t < t^*(\mu)} \exp\left(-\int_0^t \sum_{i=1}^n l \circ \phi(x_i, s) ds\right). \end{aligned}$$

### 1.3.2 PDMP sur $\mathbb{R}^d$ avec $d$ changeant à chaque saut, $(\check{X}_t)_t$

Ici, on ne regarde pas la population sous la forme d'une mesure ponctuelle, mais plutôt sous la forme d'un vecteur qui change de dimension à chaque saut. A chaque saut (ie à chaque division), deux cellules filles sont issues de la cellule qui s'est divisée. Donc le vecteur rassemblant le trait des cellules incrémente sa dimension de 1. On se place donc dans le formalisme de [Dav84], avec changement de mode :

$$\check{\mathbb{E}} = \bigcup_{k \geq 1} \{k\} \times \mathbb{R}^k = \{x = (n, \zeta) \mid n \in \mathbb{N}, \zeta = (x_1, \dots, x_n) \in \mathbb{R}^n\}.$$

Comme pour le modèle mesure, les éléments caractéristiques sont choisis de façon à être cohérents avec le modèle individu.

$$\begin{aligned} \check{l} &: \check{\mathbb{E}} \rightarrow \mathbb{R} \\ (n, \zeta) &\mapsto \sum_{i=1}^n l(x_i) \end{aligned}$$

$$\begin{aligned} \check{\phi}_n &: \mathbb{R}^n \times \mathbb{R}_+ \rightarrow \mathbb{R}^n \\ (\zeta, t) &\mapsto (\phi(x_1, t), \dots, \phi(x_n, t)) \end{aligned}$$

$$\begin{aligned} \check{Q}(x, A) &= \sum_{j=1}^n \frac{l(x_j)}{\sum_{i=1}^n l(x_i)} \mathbb{1}_A(x_{(j)}) \\ \text{avec } x_{(j)} &= \left(n+1, x_1, \dots, x_{j-1}, \frac{1}{2}x_j, x_{j+1}, \frac{1}{2}x_j\right) \in \mathbb{N} \times \mathbb{R}^{n+1} \end{aligned}$$

On note  $T_1 \sim \text{Exp}_x(\check{l} \circ \check{\phi}_n)$  pour dire :

$$\mathbb{P}_x(T_1 > t) = \mathbb{1}_{t < t_x^*} \exp\left(-\int_0^t \check{l}\left(n, \check{\phi}_n(\zeta, s)\right) ds\right).$$

On note  $\check{X}_t \in \check{E}$  le PDMP associé à cette construction.

### 1.3.3 PDMP augmenté du temps, $(\check{X}_t)_t$

Soit  $X$  un PDMP réel sur  $E$ , quelconque, de caractéristiques  $(\phi, l, Q)$ . Dans le chapitre 2, on va être amené à calculer une fonction gain/coût en rapport avec notre PDMP au sens suivant : on cherche à optimiser une performance de notre PDMP, représentée par cette fonction gain/coût prise sur notre PDMP. Or, on voudra pouvoir pondérer par le temps, ie plus le temps passe, plus notre coût est élevé. On regardera donc le processus augmenté du temps :

$$\check{X}_t = (X_t, t), \quad \forall t > 0.$$

Ce type d'augmentation fonctionne pour n'importe quel PDMP fini dimensionnel. On obtient également un PDMP, pour lequel on énonce ici les caractéristiques générales. Le nouvel espace d'état est donc

$$\check{E} = E \times \mathbb{R}_+.$$

Les nouvelles fonctions caractéristiques, notées  $(\check{l}, \check{\phi}, \check{Q})$  s'expriment pour tout  $(x, t) \in \check{E}$ , pour tout  $0 \leq s \leq t^*(x)$  et pour tout  $A$  borélien sur  $E$  :

$$\begin{aligned} \check{l}((x, t)) &= \lambda(x), \\ \check{\phi}((x, t), s) &= (\Phi(x, s), t + s), \\ \check{Q}((x, t), A \times \{t\}) &= Q(x, A), \\ \check{t}^*(x, t) &= \inf\{s > 0 \mid \check{\phi}((x, t), s) \in \partial\check{E}\} = t^*(x). \end{aligned}$$

Pour plus de détails, on pourra se référer à [Dav93], au paragraphe 31.

On peut également identifier ce PDMP augmenté du temps à un PDMP à valeur mesure. En effet, si on considère la mesure  $\mu = \sum_i \delta_{x_i}$  de  $\mathfrak{N}$  et un instant  $t > 0$ , on peut regarder la mesure  $\mu_t = \sum_i \delta_{(x_i, t)}$  qui est bien aussi une mesure de  $\mathfrak{N}$ . Comme le PDMP augmenté est aussi un PDMP réel, on peut construire un PDMP à valeur mesure comme en 1.3.1. On note  $\check{X}_t$  : le processus défini sur  $\mathfrak{N}$  avec comme point initial  $\mu_0 = \sum_{i=1}^n \delta_{(x_i, 0)}$ , et dont les éléments caractéristiques sont donnés par :

$$\begin{aligned} \check{\Phi} : \mathfrak{N} \times \mathbb{R}_+ &\rightarrow \mathfrak{N} \\ (\mu_u, t) &\mapsto \sum_{i=1}^n \delta_{(\phi(x_i, t), u+t)}, \end{aligned}$$

pour tout  $\mu_u$  in  $\mathfrak{N}$  s'écrivant comme  $\mu_u = \sum_{i=1}^n \delta_{(x_i, u)}$ ,  $u \geq 0$ . L'intensité de saut est :

$$\begin{aligned} \check{\lambda} : \mathfrak{N} &\rightarrow \mathbb{R} \\ \mu_u = \sum_{i=1}^n \delta_{(x_i, u)} &\mapsto \sum_{i=1}^n l(x_i), \end{aligned}$$

et le noyau de transition s'exprime de la façon suivante

$$\tilde{Q} \left( \sum_{i=1}^n \delta_{(x_i, u)}, A \right) = \sum_{j=1}^n \frac{l(x_j)}{\sum_{i=1}^n l(x_i)} \mathbb{1}_A \left( \sum_{i=1}^n \delta_{(x_i, u)} - \delta_{(x_j, u)} + 2\delta_{(x_j/2, u)} \right).$$

### 1.3.4 Cellule marquée et lien avec les PDMP mesure

On explique ici le modèle d'une particule marquée, dont le comportement est similaire au modèle individuel d'une cellule  $(\phi, l, \mathbf{Q})$ , avec ici une division par 2 des cellules ( $\mathbf{Q}(x, \cdot) = \delta_{x/2}(\cdot)$ ) et toujours un taux de croissance  $r$  constant (voir l'exemple 1.1.7). Ce modèle est un PDMP réel (en dimension 1) et on peut le relier directement à une dynamique de populations macroscopique. En effet, comme nous le verrons dans les théorèmes 1.3.2 et 1.3.3, il existe des formules reliant individu marqué et population totale, population totale que nous pouvons elle représenter avec un PDMP à valeurs mesures.

Ce modèle, présenté plus précisément dans [RHK<sup>+</sup>14] par exemple, nous permet d'avoir des formules reliant le comportement de la population totale de cellules avec cette particule marquée. On le présente ici, et il y sera fait référence dans le chapitre 2, où on s'intéressera à la comparaison des différentes fonctions valeurs. C'est un modèle qu'on peut rapprocher du modèle individu, où l'on reste en dimension 1, mais sans pour autant suivre une lignée de façon trajectorielle.

La construction précise du modèle "particule marquée" sera présentée dans la démonstration du théorème 1.3.2, car cela nécessite l'introduction d'autres notations (présentées dans [RHK<sup>+</sup>14]), non nécessaires à l'énoncé du-dit théorème. En effet, la particule marquée est un modèle en dimension 1 où à chaque génération  $k$ , on choisit de façon aléatoire la particule marquée parmi les cellules vivantes de cette génération  $k$ . On a donc besoin d'un modèle où l'on peut connaître la généalogie de chaque cellule. Ce qui n'est pas le cas de notre modélisation de population par des PDMP-mesures : en effet, à un instant  $t$ , notre processus  $X_t$  s'écrit comme une mesure de  $E$

$$X_t = \sum_{i=1}^{N_t} \delta_{x_i(t)}.$$

Chaque  $x_i(t)$  représente le trait d'une cellule vivante à l'instant  $t$ , mais on ne sait rien de sa généalogie. La variable  $N_t$  est le nombre de cellules vivantes à l'instant  $t$ . On a besoin d'un modèle plus fin pour pouvoir suivre des lignées.

On énonce ici la formule many-to-one, issue de [RHK<sup>+</sup>14], mais énoncée avec notre formalisme mesure :

**Théorème 1.3.2.**

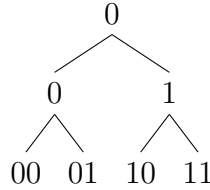
Soit  $f$  et  $h$  des fonctions mesurable bornée sur  $\mathbb{R}$ . Soit  $(\mathcal{X}_t)_t$  un PDMP de dimension 1, de caractéristiques  $(\phi, l, \mathbf{Q})$  correspondant à la division cellulaire symétrique. Soit  $(X_t)_t$  un PDMP mesure d'éléments caractéristiques construits à partir de  $(\Phi, \lambda, Q)$  comme à la sous-partie 1.3.1. Pour  $x$  dans  $\mathbb{R}$  un point initial, on a :

$$\mathbb{E}_x[f(\mathcal{X}_t)] = \mathbb{E}_{\delta_x} \left[ \frac{e^{-rt}}{x} \int_{\mathbb{R}} u f(u) X_t(du) \right], \quad (1.3.4)$$

$$\mathbb{E}_{\delta_x} \left[ e^{-rt} \int_{\mathbb{R}} h(u) X_t(du) \right] = \mathbb{E}_x \left[ x \frac{h(\mathcal{X}_t)}{\mathcal{X}_t} \right]. \quad (1.3.5)$$

**Preuve.** On commence par donner les notations de [RHK<sup>+</sup>14], qui permettent de suivre la généalogie des cellules. Dans le processus PDMP réel,  $\mathbf{X}_t$  représente la valeur du trait qui nous intéresse (taille, masse, âge, etc) à l'instant  $t$ . De plus, la quantité  $\mathcal{X}_t$  représente la valeur de ce trait pour la particule marquée, ainsi que  $\mathcal{Z}_n := \mathcal{X}_{T_n}$ . Mais comment "choisir" cette particule marquée parmi la population de cellules entière ? On procède en loi (et non pas de manière trajectorielle).

On note  $\mathcal{U} = \bigcup_{k \geq 0} \{0, 1\}^k$  l'ensemble des "cellules". On peut le représenté par un arbre, ici par exemple on a les générations 0,1 et 2 :



Pour toute cellule  $u \in \mathcal{U}$ , on note  $|u|$  la génération de  $u$ . Autrement dit,  $|u| = n$  si  $u = (u_1, \dots, u_n) \in \mathcal{U}$ . On note aussi pour la cellule  $u$  :

- ❖ sa date de naissance  $b_u$  ;
- ❖ sa taille initiale  $\xi_u$  ;
- ❖ sa durée de vie  $\zeta_u$  ;
- ❖ le  $i^e$  parent le long de la généalogie de  $u$   $m_u^i$  ;
- ❖ l'intervalle de temps correspond à la vie de  $u$   $I_u := [b_u, b_u + \zeta_u[$  ;
- ❖ la taille de  $u$  à l'instant  $t$  lorsque  $u$  est en vie  $\xi_t^u = \xi_u e^{r(t-b_u)} \mathbb{1}_{t \in I_u}$ .

Soit  $\vartheta_k$  la cellule marquée de la  $k^e$  génération. Pour suivre notre cellule marquée, il nous faut donc la suite  $(\vartheta_k)_{k \geq 0}$ , avec  $\vartheta_0 = \emptyset$ . On procède comme suit :

$$\mathbb{P}(\vartheta_k = u) = \frac{1}{2^k} \mathbb{1}_{|u|=k}.$$

C'est-à-dire qu'on choisit avec équiprobabilité sur les cellules de la même de génération. On note  $C_t$  le numéro de la génération de la cellule marquée au temps  $t$ , ce qui nous donne

$$b_{\vartheta_{C_t}} \leq t < b_{\vartheta_{C_t}} + \zeta_{\vartheta_{C_t}}.$$

On peut donc reformuler les équations (1.3.4) et (1.3.5) avec ces notations : soit  $f$  et  $h$  des fonctions mesurable bornée sur  $\mathbb{R}$ . Pour  $x$  dans  $\mathbb{R}$  un point initial, on a

$$\mathbb{E}_x[f(\mathcal{X}_t)] = \mathbb{E}_{\delta_x} \left[ \sum_{u \in \mathcal{U}} f(\xi_t^u) \frac{\zeta_t^u}{x} e^{-rt} \right], \quad (1.3.6)$$

$$\mathbb{E}_{\delta_x} \left[ \sum_{u \in \mathcal{U}} h(\xi_t^u) e^{-rt} \right] = \mathbb{E}_x \left[ x \frac{h(\mathcal{X}_t)}{\mathcal{X}_t} \right]. \quad (1.3.7)$$

On reprend la démonstration telle que faite dans [RHK<sup>+</sup>14], mais en la simplifiant pour notre exemple précis ; en effet, elle est faite dans cet article pour un taux de croissance non identique pour chaque cellule.

On fait la somme sur  $\mathcal{U}$  des évènements  $\{t \in I_u\} \cap \{u = \vartheta_{C_t}\}$ , ce qui correspond à «  $u$  est en vie en  $t$  ET  $u$  est la cellule marquée en  $t$  ». Ces évènements forment bien une famille totale.

On introduit la filtration discrète  $\mathcal{H}_n$  engendrée par  $(\xi_u, \zeta_u)$  pour tout  $u$  telle que  $|u| \leq n$ . On a alors sur  $\{t \in I_v\}$  :

$$\mathbb{P}(\vartheta_{C_t} = v | \mathcal{H}_{|v|}) = \frac{1}{2^{|v|}}.$$

De plus, sur  $\{t \in I_u\} \cap \{u = \vartheta_{C_t}\}$ , on a  $\mathcal{X}_t = \xi_t^u$ . Mais

$$\begin{aligned} \xi_t^u &= \xi_u e^{r(t-b_u)} \mathbf{1}_{t \in I_u} = \left( \frac{1}{2} \xi_{m_u^1} e^{r(b_u - b_{m_u^1})} \right) e^{r(t-b_u)} \mathbf{1}_{t \in I_u} \\ &= \left( \frac{1}{2} \left( \frac{1}{2} \xi_{m_u^2} e^{r(b_{m_u^1} - b_{m_u^2})} \right) e^{r(b_u - b_{m_u^1})} \right) e^{r(t-b_u)} \mathbf{1}_{t \in I_u} \\ &= \frac{1}{2^{|u|}} x e^{rt} \mathbf{1}_{t \in I_u}. \end{aligned}$$

D'où

$$\mathbb{P}(\vartheta_{C_t} = v | \mathcal{H}_{|v|}) = \frac{1}{2^{|v|}} = \frac{\xi_t^u}{x e^{rt}}. \quad (1.3.8)$$

Donc :

$$\begin{aligned} \mathbb{E}_x[f(\mathcal{X}_t)] &= \mathbb{E}_{\delta_x} \left[ \sum_{u \in \mathcal{U}} f(\xi_u e^{r(t-b_u)}) \mathbf{1}_{\{t \in I_u\} \cap \{u = \vartheta_{C_t}\}} \right] \\ &= \mathbb{E}_{\delta_x} \left[ \mathbb{E}_{\delta_x} \left[ \sum_{u \in \mathcal{U}} f(\xi_u e^{r(t-b_u)}) \mathbf{1}_{\{t \in I_u\} \cap \{u = \vartheta_{C_t}\}} \middle| \mathcal{H}_{|u|} \right] \right] \\ &= \mathbb{E}_{\delta_x} \left[ \sum_{u \in \mathcal{U}} f(\xi_u e^{r(t-b_u)}) \mathbf{1}_{t \in I_u} \mathbb{P}(u = \vartheta_{C_t} | \mathcal{H}_{|u|}) \right] \\ &= \mathbb{E}_{\delta_x} \left[ \sum_{u \in \mathcal{U}} f(\xi_u e^{r(t-b_u)}) \mathbf{1}_{t \in I_u} \frac{\xi_t^u}{x e^{rt}} \right] \\ &= \mathbb{E}_{\delta_x} \left[ \sum_{u \in \mathcal{U}} f(\xi_t^u) \frac{\zeta_t^u}{x} e^{-rt} \right]. \end{aligned}$$

Pour 1.3.7, il suffit de reprendre 1.3.6 avec  $f(z) = x \times h(z)/z$ .

□

Pour la suite, on va avoir besoin de many-to-one mais en partant d'un point initial à valeur mesure : ceci traduit une population initiale composée de plus d'un individu. On considère  $n$  cellules initiales, ayant chacune une taille  $x_i$ . On a alors  $X_0 = \sum_{i=1}^n \delta_{x_i}$ . On note  $\mu$  cette mesure initiale. Se pose alors le choix de la valeur initiale pour la particule marquée. On choisit uniformément sur les  $n$  cellules :  $\mathcal{X}_0 \sim \mathcal{U}(\{x_1, \dots, x_n\})$ .

**Théorème 1.3.3.**

Soit  $h$  une fonction mesurable bornée sur  $\mathbb{R}$ . En généralisant à un point de départ mesure  $\mu = \sum_{i=1}^n \delta_{x_i}$ , on obtient la formule many-to-one suivante :

$$\mathbb{E}_\mu \left[ \sum_{u \in \mathcal{U}} h(\xi_t^u) e^{-rt} \right] = \sum_{i=1}^n \mathbb{E}_{x_i} \left[ \frac{h(\mathcal{X}_t)}{\mathcal{X}_t} \right] \times x_i. \quad (1.3.9)$$

**Preuve.** La démonstration s'effectue comme précédemment, en faisant une somme sur les cellules initiales.

□

## Conclusion

Dans ce premier chapitre nous avons présenté les PDMP réels issus de la construction de Davis. Nous avons également vu notre construction de PDMP dans le cadre d'espaces de mesures ; pour cela, nous avons regardé la structure des espaces de mesures, permettant la construction de variables aléatoires sur de tels espaces selon une loi (ou une famille de lois) donnée(s).

De plus, tout au long du chapitre, notre exemple fil-rouge de modélisation de cellules nous permet d'adapter des modèles de PDMP réel pour suivre une unique cellule, à des modèles de PDMP-mesure pour étudier la totalité d'une population. Ainsi, on a pu voir un lien direct entre les deux types de modèles, notamment via les formules many-to-one qui présente un lien en loi entre le modèle de particule marquée et le modèle de population totale. Cependant, nous verrons au chapitre 2 que cette formule ne donne pas un lien entre les fonctions valeurs de chaque modèle pour l'optimisation.

Dans le chapitre suivant, on pose un problème d'arrêt d'optimal sur ses PDMP à valeurs mesure, en l'étudiant via des équations de programmation dynamique.



# CHAPITRE

## 2

# PROBLÈME D'ARRÊT OPTIMAL

### Plan du chapitre

---

<b>2.1 Arrêt optimal et programmation dynamique</b> . . . . .	<b>33</b>
2.1.1 Opérateur de programmation dynamique . . . . .	33
2.1.2 Famille de temps d'arrêt $\varepsilon$ -optimaux . . . . .	35
2.1.3 Caractérisation de la fonction valeur . . . . .	36
<b>2.2 Comparaison numérique entre la particule marquée et le PDMP mesure</b> . . . . .	<b>39</b>
2.2.1 Comparaison analytique des opérateurs sur $B_b(E)$ et $B_b(E)$ . . . . .	40
2.2.2 Etude théorique avec l'identité tronquée . . . . .	42
2.2.3 Comparaison avec la fonction valeur pour le modèle cellule-marquée. . . . .	45
<b>2.3 Etude de <math>g(Z_1)</math></b> . . . . .	<b>46</b>

---

Après avoir construit des PDMP sur des espaces qui nous intéressent dans le chapitre 1, nous nous intéressons aux problèmes d'arrêt optimal pour les PDMP à valeur mesure. De façon informelle, la résolution d'un problème d'arrêt optimal consiste à trouver le meilleur temps d'arrêt  $\tau$  pour stopper notre processus, dans le but de l'évaluer selon une certaine performance que l'on cherche à maximiser (ou minimiser selon le problème posé). Dans toute la suite, on suppose qu'on arrête au plus tard notre processus au bout de l'instant du  $N^{\text{ième}}$  saut  $T_N$ , pour un certain entier  $N$  fixé. On parle alors d'horizon aléatoire, car les temps d'arrêt admissibles ne doivent pas dépasser cette borne temporelle (aléatoire).



Nous formaliserons ces problèmes dans la première partie de ce chapitre, l'idée va être de s'intéresser à la quantité suivante en se donnant un  $\mathbf{x}$  dans  $E$  :

$$\mathbb{V}(\mathbf{x}) := \sup_{S \leq T_N} \mathbb{E}[g(X_S) \mid X_0 = \mathbf{x}], \quad (2.0.1)$$

où le suprémum se fait sur les temps d'arrêt bornés par  $T_N$ . Cette fonction  $\mathbb{V}$  est appelée *fonction valeur* et correspond à la meilleur performance moyenne avant l'instant du  $N^{\text{ième}}$  saut. Elle dépend du point initial (déterministe)  $\mathbf{x} \in E$  du processus. On rappelle que  $\mathbf{x}$  s'écrit  $(v, \mu)$  où  $v \in K$  est le mode discret et  $\mu \in \mathfrak{N}$  est une mesure. La fonction  $g$  de la formule (2.0.1) se nomme *fonction récompense*. Elle est toujours prise dans  $B_b(E)$ , l'ensemble des fonctions à valeurs réelles mesurable bornée définie en 1.2.1.

Les problèmes d'arrêt optimaux pour les PDMP ont été étudiés dans, par exemple, [CD88, Dav93, Gat92, Gug86], toujours le cas de PDMP réel, en dimension finie. C'est d'ailleurs de ce dernier article [Gug86] que nous nous inspirons pour étudier la fonction valeur et tenter de construire une suite de temps d'arrêt  $(\tau_\varepsilon)_{\varepsilon > 0}$ , bornée par  $T_N$ , vérifiant :

$$\mathbb{V}(\mathbf{x}) - \varepsilon \leq \mathbb{E}[g(X_{\tau_\varepsilon}) \mid X_0 = \mathbf{x}] \leq \mathbb{V}(\mathbf{x}).$$

On dirait alors que le temps d'arrêt  $\tau_\varepsilon$  est  $\varepsilon$ -optimal pour ce problème.

De plus, il est vu dans ces ouvrages que la fonction valeur peut être obtenue de manière théorique en itérant un certain opérateur, ce que nous adaptons à nos espaces de mesures dans la sous-section 2.1.3. Comme vu lors du chapitre 1, la construction de PDMP sur des espaces de mesures permet de s'affranchir des contraintes sur la dimension des espaces  $\mathbb{R}^d$ . Cette caractérisation de la fonction valeur par itération se fait à l'aide d'équations récursives appelées équations de *programmation dynamique*.

Après avoir correctement défini et démontré les équations de programmation dynamique de la fonction valeur pour nos PDMP-mesure, on s'interroge sur la possibilité suivante : une équivalence entre le problème d'arrêt optimal pour un PDMP mesure et le même problème d'arrêt optimal pour un PDMP réel bien choisi issu du PDMP mesure. En effet, lorsque le PDMP mesure est un processus de branchement, certaines de ses propriétés importantes (par exemple, les lois des grands nombres appliquée à des fonctions dépendant des individus) peuvent être obtenues en étudiant simplement une certaine lignée aléatoire bien choisie, appelée lignée marquée ou encore particule marquée. Ces équivalences sont obtenues grâce aux formules de type *many-to-one*, voir par exemple [Ber06, Clo11, DHKR15, Guy07]. On a présenté ces formules dans le chapitre précédent, en 1.3.3. Dans la section 2.2, on montre que cette propriété d'équivalence n'est pas vraie pour notre problème d'arrêt optimal. On fournit un contre-exemple sur un cas simple de division cellulaire : on obtient une différence significative entre la fonction valeur pour une cellule marquée correctement choisie et la fonction valeur de la population globale.

Suite à l'étude de ce contre exemple, nous nous sommes posé la question de la représentation graphique de la fonction valeur  $\mathbb{V} : E \rightarrow \mathbb{R}$ . En effet, pour des PDMP réel, la fonction valeur est définie sur  $E$ . Dans la plupart des cas, on peut la représenter sur  $\mathbb{R}$  ou sur  $\mathbb{R}^d$ . Mais comment représenter une fonction définie sur un espace de mesure ? De même pour les variables aléatoires des positions post-saut du PDMP mesure : comment représenter leur comportement ? Nous avons étudié le cas très simple de la division cellulaire symétrique où le point de départ est juste composés de deux cellules, et obtenu une

grande diversité d'histogrammes pour la variable  $g(Z_1)$  par des simulations. Nous avons donc trouvé le comportement théorique pour chaque sous cas présent dans notre exemple pourtant issu d'un modèle déjà simplifié.

Dans ce chapitre, on s'intéresse dans une première section 2.1 à la programmation dynamique et la preuve que la fonction valeur est solution des équations de la programmation dynamique. Dans une deuxième section 2.2, on montre que ce problème ne peut pas être résolu via des formules *many-to-one*. Enfin, dans une dernière partie 2.3, on étudie le comportement de  $g(Z_1)$ .

## 2.1 Arrêt optimal et programmation dynamique

Dans cette partie, nous présentons l'étude théorique du problème d'arrêt optimal, avec la résolution des équations de la programmation dynamique. On y présente aussi une suite de temps d'arrêt  $\varepsilon$ -optimaux.

### 2.1.1 Opérateur de programmation dynamique

On commence par introduire quelques notations qui serviront tout au long de cette section. Soit  $(X_t)_{t \geq 0}$  un PDMP à valeurs mesure sur  $(D_E[0, +\infty), \mathcal{F}, (\mathcal{F}_t)_{t \geq 0}, (\mathbb{P}_x)_{x \in E})$ , comme construit dans la partie 1.2.2. On note  $(\Phi, \lambda, Q)$  ses éléments caractéristiques. Soit  $N$  un entier strictement positif.

#### Notation 2.1.1.

- ❖ On note  $\mathcal{M}$  l'ensemble des temps d'arrêt pour la filtration  $(\mathcal{F}_t)_{t \geq 0}$ , et pour tout entier  $N > 0$  on note :

$$\mathcal{M}_N = \{\tau \in \mathcal{M}; \tau \leq T_N\} \quad (2.1.1)$$

l'ensemble des temps d'arrêt bornés par l'instant du  $N^{\text{ième}}$  saut du processus  $(X_t)_t$ .

- ❖ Pour  $w$  dans  $B_b(E)$  et  $\mathfrak{x}$  dans  $E$  et  $h$  une fonction mesurable sur  $E$  à valeurs réelles, on note

$$hQw(\mathfrak{x}) := h(\mathfrak{x}) \times Qw(\mathfrak{x}) = h(\mathfrak{x}) \int_E w(y)Q(\mathfrak{x}, dy).$$

L'hypothèse suivante est faite pour des raisons de simplifications. Elle est satisfaite dans la plupart des exemples issus de cas réaliste. On rappelle que pour tout  $\mathfrak{x} = (v, \mu)$  dans  $E$ , on définit le temps d'atteinte de sortie par :  $t^*(\mathfrak{x}) = \inf\{t > 0, \Phi_v(t, \mu) \in \partial E_v\}$ .

#### Hypothèses 2.1.2.

Le temps de sortie  $\mathfrak{x} \mapsto t^*(\mathfrak{x})$  est une fonction de  $B_b(E)$ .

On introduit maintenant les opérateurs suivants, généralisation aux espaces de mesures de ceux présentés dans [Gug86] par Gugerli. Soit  $\mathbb{H}$  et  $\mathbb{I}$  opérateurs de  $B_b(E)$  dans  $B_b(E \times \mathbb{R}_+)$  et  $\mathbb{K}$  de  $B_b(E)$  dans  $B_b(E)$  définis pour tout  $w$  dans  $B_b(E)$ ,  $\mathbf{x}$  dans  $E$  et  $t \geq 0$  par

$$\begin{aligned}\mathbb{H}w(\mathbf{x}, t) &= w(\Phi(\mathbf{x}, t \wedge t^*(\mathbf{x}))) e^{-\Lambda(\mathbf{x}, t \wedge t^*(\mathbf{x}))}, \\ \mathbb{I}w(\mathbf{x}, t) &= \int_0^{t \wedge t^*(\mathbf{x})} \lambda Qw(\Phi(\mathbf{x}, s)) e^{-\Lambda(\mathbf{x}, s)} ds, \\ \mathbb{K}w(\mathbf{x}) &= \int_0^{t^*(\mathbf{x})} \lambda Qw(\Phi(\mathbf{x}, s)) e^{-\Lambda(\mathbf{x}, s)} ds + Qw(\Phi(\mathbf{x}, t^*(\mathbf{x}))) e^{-\Lambda(\mathbf{x}, t^*(\mathbf{x}))},\end{aligned}\tag{2.1.2}$$

avec

$$\Lambda(\mathbf{x}, t) = \int_0^t \lambda(\Phi(\mathbf{x}, s)) ds.$$

Il est assez aisé de voir qu'on peut interpréter ces opérateurs en terme d'espérance faisant intervenir le premier instant de saut  $T_1$  et sa position associée  $Z_1 := X_{T_1}$ .

**Proposition 2.1.3.**

Pour tout  $w$  dans  $B_b(E)$ ,  $x$  dans  $E$  et  $t \geq 0$  on a :

- ❖  $\mathbb{H}w(\mathbf{x}, t) = \mathbb{E}_{\mathbf{x}} [w(X_{t \wedge t^*(\mathbf{x})}) \mathbb{1}_{S_1 > t \wedge t^*(\mathbf{x})}] = w(\Phi(\mathbf{x}, t \wedge t^*(\mathbf{x}))) \mathbb{P}_x(S_1 > t \wedge t^*(\mathbf{x})),$
- ❖  $\mathbb{I}w(\mathbf{x}, t) = \mathbb{E}_{\mathbf{x}} [w(Z_1) \mathbb{1}_{S_1 \leq t \wedge t^*(\mathbf{x})}],$
- ❖  $\mathbb{K}w(\mathbf{x}) = \mathbb{E}_{\mathbf{x}} [w(Z_1)].$

Comme dit dans l'introduction, on se donne une fonction  $g$  dans  $B_b(E)$ , la fonction récompense. On note également  $\mathbb{J}$  and  $\mathbb{L}$  les opérateurs de  $B_b(E)$  dans  $B_b(E \times \mathbb{R}_+)$  et  $B_b(E)$  respectivement définis pour tout  $w$  dans  $B_b(E)$ ,  $x$  dans  $E$  et  $t \in \mathbb{R}_+$  par :

$$\begin{aligned}\mathbb{J}w(\mathbf{x}, t) &= \mathbb{H}g(\mathbf{x}, t) + \mathbb{I}w(\mathbf{x}, t), \\ \mathbb{L}w(\mathbf{x}) &= \sup_{t \geq 0} \{ \mathbb{J}w(\mathbf{x}, t) \} \vee \mathbb{K}w(\mathbf{x}),\end{aligned}\tag{2.1.3}$$

On peut interpréter l'opérateur  $\mathbb{L}$  pris en la fonction récompense comme une maximisation faisant intervenir notre processus : on regarde d'abord le gain le plus élevée entre l'instant 0 et  $t^*(\mathbf{x})$  le long du flot, correspondant au terme  $\sup_{t \geq 0} \{ \mathbb{J}g(\mathbf{x}, t) \}$ . Puis on le compare au gain obtenu en attendant le prochain saut, représenté par le terme  $\mathbb{K}g(\mathbf{x})$ . Le tout revient à choisir la valeur la plus élevée entre les deux, ce qu'on peut encore exprimer comme "attendre le prochain saut OU maximiser la fonction le long du flot", si on s'arrête au bout d'un saut. Informellement, par la propriété de Markov si on sait optimiser sur un horizon aléatoire d'un saut, on peut espérer savoir optimiser sur un horizon de  $N$  sauts par itération.

De plus, on note pour tout  $x$  dans  $E$  et pour tout  $n$  entier supérieur à 1 :

$$\begin{cases} \mathbb{V}_0(\mathbf{x}) &= g(\mathbf{x}), \\ \mathbb{V}_n(\mathbf{x}) &= \mathbb{L}\mathbb{V}_{n-1}(\mathbf{x}). \end{cases}\tag{2.1.4}$$

Au vu de la construction des opérateurs et de la fonction récompense  $g$ , les fonctions  $\mathbb{W}_n$  sont dans  $B_b(E)$ .

Avant de montrer que la fonction valeur est solution des équations de la programmation dynamique dans la partie 2.1.3, on étudie une famille de temps d'arrêt pour lesquels on va montrer qu'ils sont  $\varepsilon$ -optimaux.

### 2.1.2 Famille de temps d'arrêt $\varepsilon$ -optimaux

On introduit maintenant un ensemble de variables aléatoires qui feront office de suite  $\varepsilon$ -optimale pour le problème d'arrêt optimal. On doit pour cela montrer que ces variables aléatoires sont des temps d'arrêt, ce qu'on fait dans cette sous section.

Pour tout  $\mathfrak{x}$  dans  $E$ ,  $n$  dans  $\mathbb{N}$  et  $\varepsilon > 0$ , on pose

$$r_{n,\varepsilon}(\mathfrak{x}) = \begin{cases} t^*(\mathfrak{x}) \text{ si } \mathbb{K}\mathbb{V}_n(\mathfrak{x}) > \sup_{t>0} \mathbb{J}\mathbb{V}_n(\mathfrak{x}, t), \\ \inf\{s \geq 0 ; \mathbb{J}\mathbb{V}_n(\mathfrak{x}, s) \geq \sup_{t>0} \mathbb{J}\mathbb{V}_n(\mathfrak{x}, t) - \varepsilon\} & \text{sinon.} \end{cases}$$

La valeur de  $r_{n,\varepsilon}(\mathfrak{x})$  dépend de qui "gagne" le sup dans  $\mathbb{L}\mathbb{V}_n(\mathfrak{x})$ , (voir (2.1.3)).

De plus, on définit par récurrence pour  $n \geq 2$

$$\begin{aligned} R_{1,0}^\varepsilon &= r_{0,\varepsilon}(Z_0), \\ R_{n,0}^\varepsilon &= r_{n-1,\varepsilon/2}(Z_0), \\ R_{n,k}^\varepsilon &= r_{n-k-1,\varepsilon/2^k}(Z_k) \mathbf{1}_{(R_{n,k-1}^\varepsilon \geq S_k)}, \quad 1 \leq k \leq n-1. \end{aligned}$$

On remarque que la variable aléatoire  $R_{n,k}^\varepsilon$  est  $\mathcal{F}_{T_k}$ -mesurable pour  $0 \leq k \leq n-1$ . Enfin, on pose :  $S_{1,\varepsilon} := r_{0,\varepsilon}(Z_0) \wedge T_1 = R_{1,0}^\varepsilon \wedge S_1$ , et par itération,

$$S_{n,\varepsilon} := \begin{cases} R_{n,0}^\varepsilon & T_1 > R_{n,0}^\varepsilon, \\ T_1 + \theta(T_1)S_{n-1,\varepsilon/2} & T_1 \leq R_{n,0}^\varepsilon, \end{cases}$$

où  $\theta(t)$  est l'opérateur de décalage de longueur  $t$  sur  $D_E[0, +\infty)$ , autrement dit pour  $f \in D_E[0, +\infty)$ ,  $\theta(t)f(\cdot) = f(t + \cdot)$ .

Pour prouver que les  $S_{n,\varepsilon}$  sont des temps d'arrêt dans  $\mathcal{M}_n$ , on commence par démontrer des propriétés sur les  $R_{n,k}^\varepsilon$  composés avec l'opérateur de décalage.

#### Lemme 2.1.4.

Pour tout  $\varepsilon > 0$  et  $n \geq 2$  et  $1 \leq k \leq n-1$ , sur l'évènement  $\{T_1 \leq S_{n,2\varepsilon}\}$ , on a :

$$R_{n,k}^{2\varepsilon} = \theta(T_1)(R_{n-1,k-1}^\varepsilon).$$

**Preuve.** Pour  $n = 2$ , par définition, on a  $R_{1,0}^\varepsilon = r_{0,\varepsilon}(Z_0)$  donc

$$\theta(T_1)(R_{1,0}^\varepsilon) = r_{0,\varepsilon}(Z_1),$$

et  $R_{2,1}^{2\varepsilon} = r_{0,\varepsilon}(Z_1)$  sur  $\{T_1 \leq S_{2,2\varepsilon}\} = \{T_1 \leq R_{2,0}^\varepsilon\}$  d'où le résultat.

Pour  $n \geq 3$  on prouve le résultat par récurrence sur  $k$  en utilisant des arguments similaires et le fait que  $\{T_1 \leq S_{n,\varepsilon}\} = \{R_{n,0}^{2\varepsilon} \geq T_1\}$ .  $\square$

On démontre maintenant que  $S_{n,\varepsilon}$  est un temps d'arrêt en utilisant la caractérisation du théorème 1.2.7, p 20.

**Lemme 2.1.5.**

Pour tout  $\varepsilon > 0$  et  $n \in \mathbb{N}^*$  on a

$$S_{n,\varepsilon} = \sum_{k=1}^n R_{n,k-1}^\varepsilon \wedge S_k.$$

En particulier,  $S_{n,\varepsilon}$  est un temps d'arrêt et  $S_{n,\varepsilon} \leq T_n$ .

**Preuve.** La démonstration se fait par récurrence sur  $n$  :

pour  $n = 1$ , par définition on a  $S_{1,\varepsilon} = r_{0,\varepsilon}(Z_0) \wedge T_1 = R_{1,k-1}^\varepsilon \wedge S_1$  et donc le résultat.

Supposons que la propriété est vraie au rang  $n - 1$ . Par définition, sur l'évènement  $\{r_{n-1,\varepsilon/2}(Z_0) = R_{n,0}^\varepsilon < S_1\}$ , on a  $S_{n,\varepsilon} = R_{n,0}^\varepsilon$  et  $R_{n,1}^\varepsilon = 0 < S_2$ ,  $R_{n,2}^\varepsilon = 0 < S_3$ , et ainsi de suite. Donc on en déduit  $\sum_{k=1}^n R_{n,k-1}^\varepsilon \wedge S_k = R_{n,0}^\varepsilon$  et ainsi la propriété est vraie au rang  $n$ .

Sur l'évènement  $\{R_{n,0}^\varepsilon \geq S_1\}$ , par définition on a  $S_{n,\varepsilon} = T_1 + \theta(T_1)S_{n-1,\varepsilon/2}$ . En combinant la récurrence et le lemme précédent 2.1.4, on a :

$$T_1 + \theta(T_1)S_{n-1,\varepsilon/2} = S_1 + \sum_{k=1}^{n-1} R_{n,k}^\varepsilon \wedge S_{k+1} = \sum_{k=1}^n R_{n,k-1}^\varepsilon \wedge S_k,$$

sur  $\{R_{n,0}^\varepsilon \geq S_1\}$ . D'où le résultat. □

### 2.1.3 Caractérisation de la fonction valeur

On reprend les équations de la programmation, définies en (2.1.4) :

$$\begin{cases} \mathbb{V}_0(\mathfrak{x}) &= g(\mathfrak{x}), \\ \mathbb{V}_n(\mathfrak{x}) &= \mathbb{L}\mathbb{V}_{n-1}(\mathfrak{x}). \end{cases}$$

On énonce ici notre résultat principal, à savoir que la fonction valeur  $\mathbb{V}$  définie en (2.0.1) par

$$\mathbb{V}(\mathfrak{x}) := \sup_{S \leq T_N} \mathbb{E}[g(X_S) \mid X_0 = \mathfrak{x}] = \sup_{S \leq T_N} \mathbb{E}_{\mathfrak{x}}[g(X_S)]$$

est égale à la  $N^{\text{ième}}$  itération de la programmation dynamique, et que  $S_{N,\varepsilon}$  est un temps d'arrêt  $\varepsilon$ -optimal.

**Théorème 2.1.6.**

Soit  $\mathfrak{x} \in E$  et  $N \in \mathbb{N}$ . Alors pour tout  $\varepsilon > 0$ ,  $S_{N,\varepsilon}$  est dans  $\mathcal{M}_N$  et :

$$\mathbb{V}_N(\mathfrak{x}) = \mathbb{V}(\mathfrak{x}) := \sup_{S \leq T_N} \mathbb{E}_{\mathfrak{x}}[g(X_S)], \quad (2.1.5)$$

$$\forall \varepsilon > 0, \quad \mathbb{E}_{\mathfrak{x}}[g(X_{S_{N,\varepsilon}})] \geq \mathbb{V}(\mathfrak{x}) - \varepsilon. \quad (2.1.6)$$

**Preuve.** Supposons dans un premier temps que l'on ait la propriété (2.1.6). Montrons alors la propriété

$$\forall S \in \mathcal{M}, \quad \forall n \in \mathbb{N}, \quad \mathbb{E}_{\mathfrak{F}}[g(X_{S \wedge T_n})] \leq \mathbb{V}_n(\mathfrak{F}). \quad (2.1.7)$$

Ces deux équations (2.1.6)+(2.1.7) permettent alors d'avoir (2.1.5). En effet, supposons (2.1.6) et (2.1.7). On a  $S \wedge T_n \in \mathcal{M}_n$  pour tout  $S \in \mathcal{M}$ . Par (2.1.7), pour tout  $n \in \mathbb{N}$ ,  $\mathfrak{F} \in \mathfrak{R}$  :

$$\sup_{S \in \mathcal{M}_n} \mathbb{E}_{\mathfrak{F}}[g(X_{S \wedge T_n})] = \sup_{S \in \mathcal{M}_n} \mathbb{E}_{\mathfrak{F}}[g(X_S)] \leq \mathbb{V}_n(\mathfrak{F}),$$

et par (2.1.6), on a l'encadrement suivant, valable pour tout  $\varepsilon > 0$

$$\mathbb{V}_n(\mathfrak{F}) - \varepsilon \leq \mathbb{E}_{\mathfrak{F}}[g(X_{S_{n,\varepsilon}})] \leq \sup_{S \in \mathcal{M}_n} \mathbb{E}_{\mathfrak{F}}[g(X_S)] \leq \mathbb{V}_n(\mathfrak{F}).$$

On en déduit donc (2.1.5). On montre maintenant donc par récurrence sur  $n$  les équations (2.1.6) et (2.1.7).

**Cas  $n=1$  :** Ce cas est basé sur le théorème 1.2.7 qui permet d'obtenir l'égalité suivante :

$$\mathbb{E}_{\mathfrak{F}}[g(X_{S \wedge T_1})] = \mathbb{E}_{\mathfrak{F}}[g(X_{R_0 \wedge T_1})],$$

pour  $R_0$  une variable aléatoire  $\mathcal{F}_0$ -mesurable. On en déduit :

$$\begin{aligned} \mathbb{E}_{\mathfrak{F}}[g(X_{S \wedge T_1})] &= \mathbb{E}_{\mathfrak{F}}[g(X_{R_0})\mathbb{1}_{T_1 > R_0}] + \mathbb{E}_{\mathfrak{F}}[g(Z_1)\mathbb{1}_{T_1 \leq R_0}] \\ &= \mathbb{H}g(\mathfrak{F}, R_0) + \mathbb{I}g(\mathfrak{F}, R_0) = \mathbb{J}g(\mathfrak{F}, R_0) \\ &\leq \sup_{t \geq 0} \{\mathbb{J}g(\mathfrak{F}, t)\} \\ &\leq \mathbb{V}_1(\mathfrak{F}), \end{aligned}$$

ce qui prouve l'initialisation de (2.1.7). Pour celle de (2.1.6), on veut montrer que

$$\forall \varepsilon > 0, \quad \mathbb{E}_{\mathfrak{F}}[g(X_{S_{1,\varepsilon}})] \geq \mathbb{V}_1(\mathfrak{F}) - \varepsilon,$$

avec par définition

$$S_{1,\varepsilon} = r_{0,\varepsilon}(Z_0) \wedge T_1 = R_{1,0}^\varepsilon \wedge T_1.$$

On regarde les différents cas séparément.

❖ Si  $\mathbb{K}g(\mathfrak{F}) > \sup_{t \geq 0} \mathbb{J}g(t, \mathfrak{F})$ , alors  $\mathbb{V}_1(\mathfrak{F}) = \mathbb{K}g(\mathfrak{F}) = \mathbb{E}_{\mathfrak{F}}[g(Z_1)]$  et aussi

$$S_{1,\varepsilon} = t^*(\mathfrak{F}) \wedge T_1 = T_1,$$

ce qui implique

$$\mathbb{E}_{\mathfrak{F}}[g(X_{S_{1,\varepsilon}})] = \mathbb{E}_{\mathfrak{F}}[g(Z_1)] = \mathbb{V}_1(\mathfrak{F}) \geq \mathbb{V}_1(\mathfrak{F}) - \varepsilon.$$

❖ Sinon,  $\mathbb{V}_1(\mathfrak{x}) = \sup_{t \geq 0} \mathbb{J}g(t, \mathfrak{x})$  et de plus

$$\mathbb{E}_{\mathfrak{x}}[g(X_{S_{1,\varepsilon}})] = \mathbb{E}_{\mathfrak{x}}[g(X_{r_{0,\varepsilon}(\mathfrak{x}) \wedge T_1})] = \mathbb{E}_{\mathfrak{x}}[g(X_r)\mathbf{1}_{T_1 \geq r}] + \mathbb{E}_{\mathfrak{x}}[g(Z_1)\mathbf{1}_{T_1 \leq r}] = \mathbb{J}g(r_{0,\varepsilon}(\mathfrak{x}), \mathfrak{x})$$

par définition de  $\mathbb{J}$ . On en déduit, par définition de  $r_{0,\varepsilon}$  :

$$\mathbb{E}_{\mathfrak{x}}[g(X_{S_{1,\varepsilon}})] \geq \sup_{t \geq 0} \{\mathbb{J}g(t, \mathfrak{x})\} - \varepsilon = \mathbb{V}_1(\mathfrak{x}) - \varepsilon.$$

Ceci termine le cas  $n = 1$  pour (2.1.6) et (2.1.7).

**Cas  $n > 1$  :** Supposons (2.1.7) soit vraie jusqu'au rang  $n$  et montrons que  $n + 1$  est vrai. La propriété au rang  $n$  s'écrit :

$$\forall S \in \mathcal{M}, \quad \mathbb{E}_{\mathfrak{x}}[g(X_{S \wedge T_n})] \leq \mathbb{V}_n(\mathfrak{x}).$$

On regarde  $S \wedge T_{n+1}$ , en faisant la disjonction  $\{S < T_1\}$  et  $\{S \geq T_1\}$  :

$$\mathbb{E}_{\mathfrak{x}}[g(X_{S \wedge T_{n+1}})] = \mathbb{E}_{\mathfrak{x}}[g(X_{S \wedge T_{n+1}})\mathbf{1}_{S < T_1}] + \mathbb{E}_{\mathfrak{x}}[g(X_{S \wedge T_{n+1}})\mathbf{1}_{S \geq T_1}].$$

Dans le premier cas,  $S \wedge T_{n+1} = S = S \wedge T_1$ . De plus, comme  $S$  est un temps d'arrêt, on peut l'écrire via le théorème 1.2.7

$$S = \sum_{n=1}^{\infty} R_{n-1} \wedge S_n,$$

où  $(R_n)_{n \in \mathbb{N}}$  sont des variables aléatoires  $(\mathcal{F}_{T_n})_{n \in \mathbb{N}}$ -adaptées. En particulier, il existe un temps d'arrêt  $S'$  dans  $\mathcal{M}$  avec lequel on peut décomposer  $S$  sur l'évènement  $\{S \geq T_1\}$  en :

$$S = S_1 + \sum_{n=2}^{\infty} R_{n-1} \wedge S_n = T_1 + \theta(T_1)(S'),$$

où  $S'$  est un temps d'arrêt dans  $\mathcal{M}$ . On a choisit de détailler ce point important de démonstration, non explicité dans [Gug86]. On en déduit alors :

$$\begin{aligned} \mathbb{E}_{\mathfrak{x}}[g(X_{S \wedge T_{n+1}})] &= \mathbb{E}_{\mathfrak{x}}[g(X_{S \wedge T_1})\mathbf{1}_{S < T_1}] + \mathbb{E}_{\mathfrak{x}}[g(X_{S \wedge T_{n+1}})\mathbf{1}_{S \geq T_1}] \\ &= \mathbb{E}_{\mathfrak{x}}[g(X_{R_0 \wedge T_1})\mathbf{1}_{R_0 < T_1}] + \mathbb{E}_{\mathfrak{x}}[g(X_{S \wedge T_{n+1}})\mathbf{1}_{R_0 \geq T_1}] \\ &= \mathbb{E}_{\mathfrak{x}}[g(X_{R_0})\mathbf{1}_{R_0 < T_1}] + \mathbb{E}_{\mathfrak{x}}[\mathbb{E}_{\mathfrak{x}}[g(X_{(T_1 + \theta(T_1)(S')) \wedge T_{n+1}})] | \mathcal{F}_1] \mathbf{1}_{R_0 \geq T_1}] \\ &= \mathbb{E}_{\mathfrak{x}}[g(X_{R_0})\mathbf{1}_{R_0 < T_1}] + \mathbb{E}_{\mathfrak{x}}[\mathbb{E}_{Z_1}[g(X_{S' \wedge T_n})]\mathbf{1}_{R_0 \geq T_1}] \tag{2.1.8} \\ &\leq \mathbb{E}_{\mathfrak{x}}[g(X_{R_0})\mathbf{1}_{R_0 < T_1}] + \mathbb{E}_{\mathfrak{x}}[\mathbb{V}_n(Z_1)\mathbf{1}_{R_0 \geq T_1}] \tag{2.1.9} \\ &\leq \sup_{r \geq 0} \mathbb{E}_{\mathfrak{x}}[g(X_r)\mathbf{1}_{r < T_1}] + \mathbb{E}_{\mathfrak{x}}[\mathbb{V}_n(Z_1)\mathbf{1}_{r \geq T_1}] \\ &\leq \sup_{r \geq 0} \mathbb{J}\mathbb{V}_n(r, \mathfrak{x}) \vee \mathbb{K}\mathbb{V}_n(\mathfrak{x}) \\ &= \mathbb{V}_{n+1}(\mathfrak{x}). \end{aligned}$$

La ligne (2.1.8) s'obtient par la propriété de Markov forte du processus et la ligne (2.1.9) se déduit de l'hypothèse de récurrence. Ceci termine la récurrence et montre bien (2.1.7). Reste à voir (2.1.6). On suppose que la propriété est vraie jusqu'au rang  $n$ . On a donc :  $\mathfrak{x}$

$$\mathbb{E}_{\mathfrak{x}}[g(X_{S_{n,\varepsilon}})] \geq \mathbb{V}_n(\mathfrak{x}) - \varepsilon.$$

On a alors :

$$\begin{aligned}
\mathbb{E}_{\mathfrak{f}}[g(X_{S_{n+1,2\varepsilon}})] &= \mathbb{E}_{\mathfrak{f}}[g(X_{S_{n+1,2\varepsilon}})\mathbb{1}_{r_{n,\varepsilon}(\mathfrak{f}) < T_1}] + \mathbb{E}_{\mathfrak{f}}[g(X_{S_{n+1,2\varepsilon}})\mathbb{1}_{r_{n,\varepsilon}(\mathfrak{f}) \geq T_1}] \\
&= \mathbb{E}_{\mathfrak{f}}[g(X_{r_{n,\varepsilon}(\mathfrak{f})})\mathbb{1}_{r_{n,\varepsilon}(\mathfrak{f}) < T_1}] + \mathbb{E}_{\mathfrak{f}}[g(X_{T_1+\theta(T_1)S_{n,\varepsilon}})\mathbb{1}_{r_{n,\varepsilon}(\mathfrak{f}) \geq T_1}] \\
&= \mathbb{E}_{\mathfrak{f}}[g(X_{r_{n,\varepsilon}(\mathfrak{f})})\mathbb{1}_{r_{n,\varepsilon}(\mathfrak{f}) < T_1}] + \mathbb{E}_{\mathfrak{f}}[\mathbb{E}_{Z_1}[g(X_{S_{n,\varepsilon}})]\mathbb{1}_{r_{n,\varepsilon}(\mathfrak{f}) \geq T_1}] \\
&\geq \mathbb{E}_{\mathfrak{f}}[g(X_{r_{n,\varepsilon}(\mathfrak{f})})\mathbb{1}_{r_{n,\varepsilon}(\mathfrak{f}) < T_1}] + \mathbb{E}_{\mathfrak{f}}[\mathbb{V}_n(Z_1)\mathbb{1}_{r_{n,\varepsilon}(\mathfrak{f}) \geq T_1}] - \varepsilon \times \mathbb{P}_{\mathfrak{f}}(r_{n,\varepsilon}(\mathfrak{f}) \geq T_1) \\
&\geq \mathbb{E}_{\mathfrak{f}}[g(X_{r_{n,\varepsilon}(\mathfrak{f})})\mathbb{1}_{r_{n,\varepsilon}(\mathfrak{f}) < T_1}] + \mathbb{E}_{\mathfrak{f}}[\mathbb{V}_n(Z_1)\mathbb{1}_{r_{n,\varepsilon}(\mathfrak{f}) \geq T_1}] - \varepsilon.
\end{aligned}$$

Or, avec la définition de  $r_{n,\varepsilon}(\mathfrak{f})$  :

- ❖ si  $\mathbb{KW}_n(\mathfrak{f}) > \sup_{t \geq 0} \{\mathbb{JW}_n(t, \mathfrak{f})\}$ , alors  $\mathbb{V}_{n+1}(\mathfrak{f}) = \mathbb{KW}_n(\mathfrak{f}) = \mathbb{E}_{\mathfrak{f}}[\mathbb{V}_n(Z_1)]$  et  $r_{n,\varepsilon}(\mathfrak{f}) = t^*(\mathfrak{f})$  donc l'inégalité continue en

$$\begin{aligned}
\mathbb{E}_{\mathfrak{f}}[g(X_{S_{n+1,2\varepsilon}})] &\geq \mathbb{E}_{\mathfrak{f}}[g(X_{t^*(\mathfrak{f})})\mathbb{1}_{t^*(\mathfrak{f}) < T_1}] + \mathbb{E}_{\mathfrak{f}}[\mathbb{V}_n(Z_1)\mathbb{1}_{T_1 \geq t^*(\mathfrak{f})}] - \varepsilon \\
&= 0 + \mathbb{KW}_n(\mathfrak{f}) - \varepsilon \\
&= \mathbb{V}_{n+1}(\mathfrak{f}) - \varepsilon.
\end{aligned}$$

- ❖ sinon,  $\mathbb{V}_{n+1}(\mathfrak{f}) = \sup_{t \geq 0} \{\mathbb{JW}_n(t, \mathfrak{f})\}$  et  $r_{n,\varepsilon}(\mathfrak{f}) = \inf\{s \geq 0 ; \mathbb{JW}_n(\mathfrak{f}, s) \geq \sup_{t > 0} \mathbb{JW}_n(\mathfrak{f}, t) - \varepsilon\}$  d'où :

$$\begin{aligned}
\mathbb{E}_{\mathfrak{f}}[g(X_{S_{n+1,2\varepsilon}})] &\geq \mathbb{JW}_n(\mathfrak{f}, r_{n,\varepsilon}(\mathfrak{f})) - \varepsilon \\
&\geq \sup_{t \geq 0} \{\mathbb{JW}_n(\mathfrak{f}, t)\} - \varepsilon - \varepsilon = \mathbb{V}_{n+1}(\mathfrak{f}) - 2\varepsilon.
\end{aligned}$$

En faisant le bilan des deux cas, on obtient bien notre inégalité (2.1.6). Ce qui termine la preuve. □

On a donc montré que les fonctions valeurs à horizon aléatoire  $T_N$  peuvent être obtenues par itération de l'opérateur  $\mathbb{L}$ . Dans l'article [Gug86], Gugerli le montrait pour des espaces réels, et nous l'avons montré pour des PDMP mesures.

Nous avons également vu dans la partie 1.3.4 du chapitre 1 qu'il existe des formules liant le comportement moyen des PDMP mesure dans le cas d'une population de cellule et des PDMP réels modélisant une cellule unique. On s'intéresse maintenant à étudier ces correspondances, et leur non-fonctionnement pour les fonctions valeurs.

## 2.2 Comparaison numérique entre la particule marquée et le PDMP mesure

On cherche ici à montrer une utilité d'introduire des nouveaux opérateurs de Gugerli pour les processus à valeurs mesures. On cherche à exprimer ces opérateurs au cas particulier de nos modèles fils rouge issus de l'exemple 1.2.3 et de la sous-section 1.3. Puis on étudie une illustration numérique des fonctions valeurs pour les PDMP mesure et les PDMP réels (dans le cas particulier de la division cellulaire symétrique).



**Table 2.1** – Comparaison des éléments caractéristiques entre PDMP réel et PDMP mesure

	PDMP réel sur $\mathbb{R}^d$	PDMP mesure sur $\mathfrak{R}$
flot	$\phi(x, t)$	$\Phi(\mu, t) = \sum_{i=1}^n \delta_{\phi(x_i, t)}$
taux de saut	$l(x)$	$\lambda(\mu) = \sum_{i=1}^n l(x_i)$
noyau	$Q(x, \mathcal{A}) = \mathbb{1}_{\mathcal{A}}(x/2)$	$Q(\mu, A) = \sum_{j=1}^m \frac{l(x_j)}{\sum_{i=1}^m l(x_i)} \times \mathbb{1}_A \left( \sum_{i \neq j} \delta_{x_i} + 2\delta_{x_j/2} \right)$

### 2.2.1 Comparaison analytique des opérateurs sur $B_b(\mathbf{E})$ et $B_b(E)$

On se place avec un mode unique. On fera donc l'abus de langage de dire que l'espace d'état est  $\mathbf{E} = \mathbb{R}$  et  $E = \mathfrak{R}$ , et aussi  $x = \zeta \in \mathbb{R}$  et  $\mathfrak{x} = \mu \in \mathfrak{R}$ . On fait ici un petit tableau récapitulatif de la partie 1.3.1, donnant les éléments caractéristiques du PDMP mesure déduits des éléments caractéristiques du PDMP réel, lorsqu'on a un noyau markovien correspondant à une division symétrique. Soient  $x, x_1, \dots, x_n$  dans  $\mathbb{R}$  et  $\mu = \sum_{i=1}^n \delta_{x_i}$  dans  $\mathfrak{R}$ . On renvoie à la table 2.1.

#### Opérateurs pour le PDMP-réel "cellule unique"

Pour  $x$  dans  $\mathbf{E}$ ,  $t \geq 0$  et  $b$  dans  $B_b(\mathbf{E})$ , les opérateurs pour le modèle individuel sont définis par :

$$\begin{aligned} \mathbf{H}b(x, t) &= b(\phi(x, t \wedge t^*(x))) \mathbb{P}_x(S_1 > t \wedge t^*(x)), \\ \mathbf{I}b(x, t) &= \mathbb{E}_x \left[ b \left( \frac{\phi(x, T_1)}{2} \right) \mathbb{1}_{S_1 \leq t \wedge t^*(x)} \right], \\ \mathbf{K}b(x) &= \mathbb{E}_x \left[ b \left( \frac{\phi(x, T_1)}{2} \right) \right]. \end{aligned}$$

#### Opérateur pour le PDMP-mesure "population"

Pour  $\mu = \sum_{i=1}^n \delta_{x_i}$  dans  $E$ ,  $t \geq 0$  et  $w$  dans  $B_b(E)$ , les opérateurs pour le modèle individu centré sont définis par :

$$\mathbf{H}w(\mu, t) = w(\Phi(\mu, t \wedge t^*(\mu))) \mathbb{P}_\mu(T_1 > t \wedge t^*(\mu)),$$

$$\begin{aligned}
\mathbb{I}w(\mu, t) &= \int_0^{t \wedge t^*(\mu)} \lambda Qw(\Phi(\mu, s)) e^{-\Lambda(\mu, s)} ds \\
&= \int_0^{t \wedge t^*(\mu)} \sum_{i=1}^n l \circ \phi(x_i, s) \times \left( \sum_{j=1}^n \frac{l \circ \phi(x_j, s)}{\sum_{k=1}^n l \circ \phi(x_k, s)} w(\Phi(\mu, s))_{(j)} \right) e^{-\Lambda(\mu, s)} ds \\
&= \sum_{i=1}^n \int_0^{t \wedge t^*(\mu)} l \circ \phi(x_i, s) w \left( \sum_{j \neq i} \delta_{\phi(x_j, s) + 2\delta_{\phi(x_i, s)/2}} \right) e^{-\Lambda(\mu, s)} ds,
\end{aligned}$$

$$\begin{aligned}
\mathbb{K}w(\mu) &= \mathbb{E}_\mu[w(Z_1)] = \mathbb{I}(\mu, t^*(\mu)) + Qw(\Phi(\mu, t^*))e^{\Lambda(\mu, t^*(\mu))} \\
&= \sum_{j=1}^n \int_0^{t^*(\mu)} l \circ \phi(x_j, s) w \left( \sum_{i \neq j} \delta_{\phi(x_i, s) + 2\delta_{\phi(x_j, s)/2}} \right) e^{-\Lambda(\mu, s)} ds \\
&\quad + \left( \sum_{j=1}^n \frac{l \circ \phi(x_j, t^*(\mu))}{\sum_{k=1}^n l \circ \phi(x_k, t^*(\mu))} w(\Phi(\mu, t^*(\mu)))_{(j)} \right) e^{-\Lambda(\mu, t^*(\mu))}.
\end{aligned}$$

Dans la suite, pour faciliter les calculs, on introduit les notations suivantes.

**Notation 2.2.1.**

On note, pour  $\mu = \sum_{i=1}^n \delta_{x_i}$  et  $t > 0$  :

$$u_j(\mu, t) := \frac{l \circ \phi(x_j, t \wedge t^*(\mu))}{\sum_{k=1}^n l \circ \phi(x_k, t \wedge t^*(\mu))} \quad \text{et} \quad U_j(\mu, t) := \prod_{i \neq j} e^{-L(x_i, t)} = \frac{e^{-\Lambda(\mu)}}{e^{-L(x_j)}}$$

avec

$$L(x, t) = \int_0^t l(\phi(x, s)) ds.$$

On cherche à exprimer les opérateurs  $\mathbb{H}, \mathbb{I}, \mathbb{L}$  en fonction de leur équivalent pour les PDMP de dimension finie,  $\mathbf{H}, \mathbf{I}, \mathbf{L}$ . Avoir une expression des uns en fonctions des autres permettrait d'utiliser les résultats sur les PDMP réels.

De plus, on suppose que les fonctions  $w$  sont de la forme  $w(\sum_i \delta_{x_i}) = \sum_i b(x_i)$  où  $b : \mathbb{R} \rightarrow \mathbb{R}$ . On choisit cette forme en vu de la comparaison numérique de la partie 2.2.2 où la fonction gain aura cette forme, en rapport avec les formules many-to-one, 1.3.3. D'où :

$$\begin{aligned}
\blacklozenge \mathbb{H}w(\mu, t) &= \left[ \sum_j b \circ \phi(x_j, t \wedge t^*(\mu)) \right] \prod_{j=1}^n e^{-L(x_j, t \wedge t^*(\mu))} \\
&= \sum_j b \circ \phi(x_j, t \wedge t^*(\mu)) e^{-L(x_j, t \wedge t^*(\mu))} U_j(\mu, t \wedge t^*(\mu)).
\end{aligned}$$

On reconnaît presque une somme de  $\mathbf{H}b(x_i, t)$ , pondérée par les  $U_j(\mu, t \wedge t^*(\mu))$ . Le "presque" vient du  $t \wedge t^*(\mu)$  : si on voulait les vrais  $\mathbf{H}b(x_i, t)$ , il faudrait avoir du  $t \wedge t^*(x_i)$

dans l'égalité ci-dessus.

$$\begin{aligned}
\blacklozenge \mathbb{I}w(\mu, t) &= \sum_{j=1}^n \int_0^{t \wedge t^*(\mu)} l \circ \phi(x_j, s) \left( \sum_{i \neq j} b \circ \phi(x_i, s) + 2b(\phi(x_j, s)/2) \right) e^{-\Lambda(\mu, s)} ds \\
&= \sum_{j=1}^n \left[ \sum_{i \neq j} \int_0^{t \wedge t^*(\mu)} l \circ \phi(x_j, s) \times b \circ \phi(x_i, s) e^{-L(x_j, s)} \times U_j(\mu, s) ds \right. \\
&\quad \left. + 2 \int_0^{t \wedge t^*(\mu)} l \circ \phi(x_j, s) \times b(\phi(x_j, s)/2) e^{-L(x_j, s)} \times U_j(\mu, s) ds \right] \\
&= \sum_{j=1}^n \left[ \sum_{i \neq j} \mathbb{E}_{x_j} [b \circ \phi(x_i, T_1) \times U_j(\mu, T_1) \mathbb{1}_{T_1 \leq t \wedge t^*}] \right. \\
&\quad \left. + 2 \mathbb{E}_{x_j} [b(\phi(x_j, T_1)/2) \times U_j(\mu, T_1) \mathbb{1}_{T_1 \leq t \wedge t^*}] \right], \\
\blacklozenge \mathbb{K}w(\mu) &= \mathbb{I}w(\mu, t^*(\mu)) + \left( \sum_{j=1}^n \frac{l \circ \phi(x_j, t^*(\mu))}{\sum_{k=1}^j l \circ (x_k, t^*(\mu))} w(\Phi(\mu, t^*(\mu)))_{(j)} \right) e^{-\Lambda(\mu, t^*(\mu))} \\
&= \mathbb{I}w(\mu, t^*(\mu)) + \sum_{j=1}^n \left( \sum_{i \neq j} u_j(\mu, t^*(\mu)) b \circ \phi(x_i, t^*(\mu)) e^{-L(x_j, t^*(\mu))} U_j(\mu, t^*(\mu)) \right. \\
&\quad \left. + 2u_j(\mu, t^*(\mu)) b(\phi(x_j, t^*(\mu))/2) e^{-L(x_j, t^*(\mu))} U_j(\mu, t^*(\mu)) \right) \\
&= \sum_{j=1}^n \sum_{i \neq j} \left[ \int_0^{t \wedge t^*(\mu)} l \circ \phi(x_j, s) \times b \circ \phi(x_i, s) e^{-L(x_j, s)} \times U_j(\mu, s) ds \right. \\
&\quad \left. + u_j(\mu, t^*(\mu)) b \circ \phi(x_i, t^*(\mu)) e^{-L(x_j, t^*(\mu))} U_j(\mu, t^*(\mu)) \right] \\
&\quad + 2 \left[ \int_0^{t \wedge t^*(\mu)} l \circ \phi(x_j, s) \times b(\phi(x_j, s)/2) e^{-L(x_j, s)} \times U_j(\mu, s) ds \right. \\
&\quad \left. + u_j(\mu, t^*(\mu)) b(\phi(x_j, t^*(\mu))/2) e^{-L(x_j, t^*(\mu))} U_j(\mu, t^*(\mu)) \right].
\end{aligned}$$

On voit ici, qu'en prenant un noyau très simple, on obtient aucune simplification évidente, ou du moins aucun lien a priori visible entre les opérateurs pour le PDMP mesure et les opérateurs pour le PDMP fini dimensionnel.

On se place sur un exemple simple afin de montrer que même dans ce cas, on observe des différences entre le cas *processus à valeurs mesures* et *cellule marquée*, présentés dans 1.3.4. On regarde pour cela les deux fonctions valeurs associées à chaque processus.

## 2.2.2 Etude théorique avec l'identité tronquée

On prend toujours  $\mu = \sum_{i=1}^n \delta_{x_i}$  une mesure ponctuelle de  $\mathfrak{X}$ . Comme dans la partie 1.3.3, on note  $\mu_t := \sum_{i=1}^n \delta_{x_i, t}$  pour  $t \geq 0$ , qui est une mesure de  $\mathfrak{X}$ . On rappelle qu'on s'intéresse à la quantité

$$\mathbb{V}(\mu) = \sup_{\tau \in \mathcal{M}_N} \mathbb{E}_\mu [g(X_\tau)],$$

où  $g$  est la fonction de performance.

Pour la fonction performance du PDMP associé à la cellule marquée, on considère la fonction de performance suivante (notée  $f$ ) :

$$\forall x \in \mathbb{R}, f(x) = \gamma + (x - \gamma)\mathbb{1}_{x < \gamma} = \begin{cases} x & \text{si } x < \gamma \\ \gamma & \text{si } x \geq \gamma \end{cases},$$

où  $\gamma > 0$  est un seuil fixé déterministe. Cela correspond à la fonction identité tronquée en  $\gamma$ . Afin de pouvoir comparer avec le processus à valeurs mesure, on utilise la formule *many-to-one* (1.3.9). La fonction de performance  $g$  correspondante est :

$$g(\mu_t) = e^{-rt} \sum_{i=1}^n f(x_i) = e^{-rt} \sum_{i=1}^n \gamma + (x_i - \gamma)\mathbb{1}_{x_i < \gamma}.$$

On a donc une pondération temporelle par rapport à la fonction  $f$ . Cela signifie que si on attend trop longtemps pour stopper notre processus, la performance va être écrasée par l'exponentielle.

On doit regarder les fonctions  $\mathbb{W}_n$  pour  $0 \leq n \leq N$ . Ces fonctions sont définies sur  $\mathfrak{R}$ , et il n'est pas possible d'en avoir une représentation graphique. Pour les PDMP réels, on obtient des fonctions définies sur  $\mathbb{R}^d$ , dont il est possible d'obtenir une discrétisation et une approximation numérique via des techniques de quantification, voir par exemple [dSDG10]. En explicitant la récurrence (2.1.4), on a :

$$\begin{aligned} \mathbb{W}_0(\mu) &= g(\mu) \\ \text{et } \mathbb{W}_{n+1}(\mu) &= \mathbb{L}\mathbb{W}_n(\mu) \\ &= \sup_{t \geq 0} \left\{ \mathbb{E}_\mu[\mathbb{W}_n(Z_1)\mathbb{1}_{T_1 \leq t}] + g \circ \Phi(\mu, t) \times \mathbb{P}_\mu(T_1 > t) \right\} \vee \mathbb{E}_\mu[\mathbb{W}_n(Z_1)]. \end{aligned}$$

En spécifiant aux mesures particulières  $Z_0, Z_1, \dots$ , on obtient une récurrence de variables aléatoires et non plus une récurrence de fonctions. Cependant, cela reste quand même des variables aléatoires à valeurs mesure, ce qui pose problème pour l'approximation numérique ; par ailleurs, la représentation graphiques (par histogramme par exemple) est compliquée. On a :

$$\begin{aligned} \mathbb{W}_0(Z_N) &= g(Z_N) \\ \mathbb{W}_1(Z_{N-1}) &= \sup_{t \geq 0} \left\{ \mathbb{E}_{Z_{N-1}}[\mathbb{W}_0(Z_1)\mathbb{1}_{T_1 \leq t}] + g \circ \Phi(Z_{N-1}, t) \times \mathbb{P}_{Z_{N-1}}(T_1 > t) \right\} \vee \mathbb{E}_{Z_{N-1}}[\mathbb{W}_0(Z_1)] \\ &= \sup_{t \geq 0} \left\{ \mathbb{E}_\mu[\mathbb{W}_0(Z_N)\mathbb{1}_{T_N \leq t}] + g \circ \Phi(Z_{N-1}, t) \times \mathbb{P}_\mu(T_N > t) \right\} \vee \mathbb{E}_\mu[\mathbb{W}_0(Z_N)] \\ &\dots \\ \mathbb{W}_N(Z_0) &= \sup_{t \geq 0} \left\{ \mathbb{E}_{Z_0}[\mathbb{W}_{N-1}(Z_1)\mathbb{1}_{T_1 \leq t}] + g \circ \Phi(Z_0, t) \times \mathbb{P}_{Z_0}(T_1 > t) \right\} \vee \mathbb{E}_{Z_0}[\mathbb{W}_{N-1}(Z_1)]. \end{aligned}$$

Comme on cherche à illustrer une différence numérique, on se place dans un cas encore plus simple. Afin de simplifier les calculs, on se place dans des conditions particulières :

### Hypothèses 2.2.2.

- ❖ (2.2.2.1) on s'arrête à l'horizon  $N = 1$  : on doit donc regarder  $\mathbb{V}_0(Z_1)$  et  $\mathbb{V}_1(Z_0)$ .
- ❖ (2.2.2.2)  $Z_0 = \mu$  est déterministe, donc  $\mathbb{V}_1(Z_0)$  aussi.
- ❖ (2.2.2.3) on se place dans le cadre de notre exemple fil rouge de division en deux cellules égales, avec une croissance exponentielle et une intensité de saut en fonction puissance.

Dans ce cas, on a :

$$\mathbb{V}_0(Z_1) = g(Z_1) = e^{-rT_1} \left[ \sum_{i \neq I_1} (x_i e^{rT_1} - \gamma) \mathbf{1}_{x_i < \gamma e^{-rT_1}} + (x_{I_1} e^{rT_1} - 2\gamma) \mathbf{1}_{x_{I_1} < 2\gamma e^{-rT_1}} + (n+1)\gamma \right]$$

où  $T_1 \sim \exp_\mu(\lambda \circ \Phi)$  et  $I_1$  est une variable aléatoire représentant le numéro de la cellule qui se divise à l'instant du saut. La loi discrète de  $I_1$  est donnée par  $\mathbb{P}(I_1 = i) = \frac{x_i^\alpha}{x_1^\alpha + \dots + x_n^\alpha}$ . On peut donc simuler la variable aléatoire  $\mathbb{V}_0(Z_1)$ , car on sait simuler  $T_1$  et  $I_1$  par leur loi explicite.

### Proposition 2.2.3.

Soit  $\gamma, r, \alpha > 0$  et  $x_1, \dots, x_n$  dans  $\mathbb{R}$ . Dans le cadre des hypothèses 2.2.2, on a :

$$\begin{aligned} \mathbb{V}_0(Z_1) &= g(Z_1) & (2.2.1) \\ &= e^{-rT_1} \left[ \sum_{i \neq I_1} (x_i e^{rT_1} - \gamma) \mathbf{1}_{x_i < \gamma e^{-rT_1}} + (x_{I_1} e^{rT_1} - 2\gamma) \mathbf{1}_{x_{I_1} < 2\gamma e^{-rT_1}} + (n+1)\gamma \right] \end{aligned}$$

et

$$\mathbb{V}_1(\mu) = \sup_{t>0} \left\{ \mathbb{E}_\mu[A_t] + b_t \right\} \vee \mathbb{E}_\mu[\mathbb{V}_0(Z_1)] \quad (2.2.2)$$

avec, pour tout  $t > 0$ ,

$$\begin{aligned} A_t &= e^{-rt} \left( \sum_{i \neq I_1} (x_i e^{rT_1} - \gamma) \mathbf{1}_{x_i < \gamma e^{-rT_1}} + (x_{I_1} e^{rT_1} - 2\gamma) \mathbf{1}_{x_{I_1} < 2\gamma e^{-rT_1}} + (n+1)\gamma \right) \mathbf{1}_{T_1 \leq t}, \\ b_t &= e^{-rt} \left( \sum_i (x_i e^{rt} - \gamma) \mathbf{1}_{x_i < \gamma e^{-rt}} + n\gamma \right) \exp \left( - \sum_i (x_i^\alpha) \frac{e^{rt\alpha} - 1}{\alpha r} \right). \end{aligned}$$

On a maintenant toutes les clés en main pour pouvoir implémenter numériquement la comparaison entre les cellules marquées.

### 2.2.3 Comparaison avec la fonction valeur pour le modèle cellule-marquée.

On peut également simuler  $\mathbb{V}_1(Z_0)$ , qui est une quantité déterministe, calculée à partir de la variable aléatoire  $\mathbb{V}_0(Z_1)$ . On explique ici par pseudo-code les algorithmes de la partie A.2.1.2, simulation de la fonction  $\mathbb{V}_1$  par la formule (2.2.2).

---

**Algorithm 2.2.1:**  $\widehat{\mathbb{V}}_1(\Delta t, nb_{max}, \mu, \gamma, r, \alpha)$

---

(Le suprémum est pris sur l'intervalle  $[0, \Delta t \times nb_{max}]$ )

$\hat{a} \leftarrow$  vecteur de taille  $nb_{max} + 1$   
 $\hat{b} \leftarrow$  vecteur de taille  $nb_{max} + 1$   
**pour tout**  $j \in \{0, 1, \dots, nb_{max}\}$   
**faire**  $\begin{cases} \hat{a}(j) \leftarrow \mathbb{E}_\mu[A_{j\Delta t}] \text{ par Monte-Carlo} \\ \hat{b}(j) \leftarrow b_{j\Delta t} \end{cases}$   
 $sup \leftarrow \max(\hat{a} + \hat{b})$   
 $v_0 \leftarrow \mathbb{E}_\mu[\mathbb{V}_0(Z_1)] \text{ par Monte-Carlo}$   
 $\widehat{\mathbb{V}}_1 \leftarrow \max\{sup, v_0\}$

---

On cherche maintenant à comparer avec la fonction valeur pour une cellule marquée. Avec le même principe, on obtient :

#### Proposition 2.2.4.

Pour tout  $x$  réel,  $\gamma, r, \alpha > 0$ , la fonction valeur pour un arrêt au premier saut pour le modèle PDMP réel  $\mathcal{V}_1$  est donnée par l'expression suivante :

$$\mathcal{V}_1(x) = \sup_{t>0} \left\{ \mathbb{E}_\mu[\mathcal{A}_t] + \beta_t \right\} \vee \mathbb{E}_x[\mathcal{V}_0(\mathcal{Z}_1)] \quad (2.2.3)$$

avec, pour tout  $t > 0$ ,

$$\begin{aligned} \mathcal{A}_t &= \left( (xe^{rT_1}/2 - \gamma) \mathbf{1}_{xe^{rT_1} < 2\gamma} + \gamma \right) \mathbf{1}_{T_1 \leq t}, \\ \beta_t &= \left[ (xe^{rt} - \gamma) \mathbf{1}_{xe^{rt} < \gamma} + \gamma \right] \exp\left( -\frac{x^\alpha}{\alpha r} (e^{r\alpha t} - 1) \right). \end{aligned}$$

Avec le même principe que précédemment, on peut également simuler  $\mathcal{V}_1(x)$ . Après simulation, on compare les deux fonctions valeurs  $\mathcal{V}_1(x)$  et  $\mathbb{V}_1(\delta_x)$  partant du même point initial. Ces deux quantités déterministes sont des nombres réels. Cependant, pour obtenir une valeur par simulation, on utilise une méthode de Monte-Carlo, ce qui introduit de l'aléatoire : plus formellement, on reprend les formules (2.2.2) et (2.2.3) ainsi que le pseudo-algorithme 2.2.1. Les quantités  $\mathcal{A}_t$  et  $A_t$  sont des variables aléatoires pour

lesquelles on peut simuler un échantillon. Afin d'avoir leur espérance, on utilise la moyenne empirique (méthode de Monte-Carlo). En réinjectant dans (2.2.2) et (2.2.3), on a donc des valeurs estimées pour  $\mathcal{V}_1(x)$  et  $\mathbb{W}_1(\delta_x)$ .

Pour obtenir les résultats ci-dessous à la table 2.2, on a simulé 100 fois  $\mathcal{V}_1(3)$  et  $\mathbb{W}_1(\delta_3)$ . Les paramètres sont :  $\alpha = 1$ ,  $r = 2$ ,  $\gamma = 1$ . Pour le pas de discrétisation  $\Delta t$ , on prend la valeur maximum des  $T_1$  (dans le Monte-Carlo), et on la divise par un nombre  $nb_{max}$  qui correspond au nombre de points de discrétisation que l'on souhaite. Ici,  $nb_{max} = 500$ . On obtient les intervalles de confiance à 95% suivants :

**Table 2.2** – Comparaison numérique entre  $\mathcal{V}_1(3)$  et  $\mathbb{W}_1(\delta_3)$

$\mathcal{V}_1(3)$	$1.0018 \pm 2.54 \times 10^{-4}$
$\mathbb{W}_1(\delta_3)$	$1.3447 \pm 3.6034 \times 10^{-4}$

Avec un point initial identique (une seule cellule), on voit que numériquement la valeur obtenue pour  $\mathcal{V}_1(3)$  et de  $\mathbb{W}_1(\delta_3)$  ne sont pas identiques, même en choisissant des fonctions récompenses compatibles au changement de modèle (individu  $\rightarrow$  population) et en prenant un horizon  $N = 1$ . D'où la nécessité d'une nouvelle étude d'optimisation pour les mesures.

## 2.3 Etude de $g(Z_1)$

Après avoir simulé la variable aléatoire  $g(Z_1) = \mathbb{W}_0(Z_1)$ , on remarque qu'on a une grande variété d'histogrammes, dont on donne un aperçu à la figure 2.3.1.

Afin d'expliquer ces différents histogrammes, on cherche la loi de  $\mathbb{W}_0(Z_1)$ . On se place dans les hypothèses 2.2.2 et avec seulement deux cellules initiales, c'est-à-dire qu'on considère  $\mu = \delta_{x_1} + \delta_{x_2}$  comme point initial du processus. On suppose de plus que  $x_1 \geq x_2$ , sans perte de généralités. On prend une fonction  $h$  continue bornée, et on étudie  $\mathbb{E}_\mu[h(\mathbb{W}_0(Z_1))]$ , selon la formule (2.2.1) explicitant  $\mathbb{W}_0(Z_1)$ . On donne avant cela les lois de  $T_1$  et  $I_1$ , par

$$\mathbb{P}(I_1 = i) = \frac{x_i^\alpha}{x_1^\alpha + x_2^\alpha}$$

et

$$f_{T_1}(t) = (x_1^\alpha + x_2^\alpha)e^{r\alpha t} \exp\left(-\frac{x_1^\alpha + x_2^\alpha}{r\alpha}(e^{r\alpha t} - 1)\right) \mathbf{1}_{t>0},$$

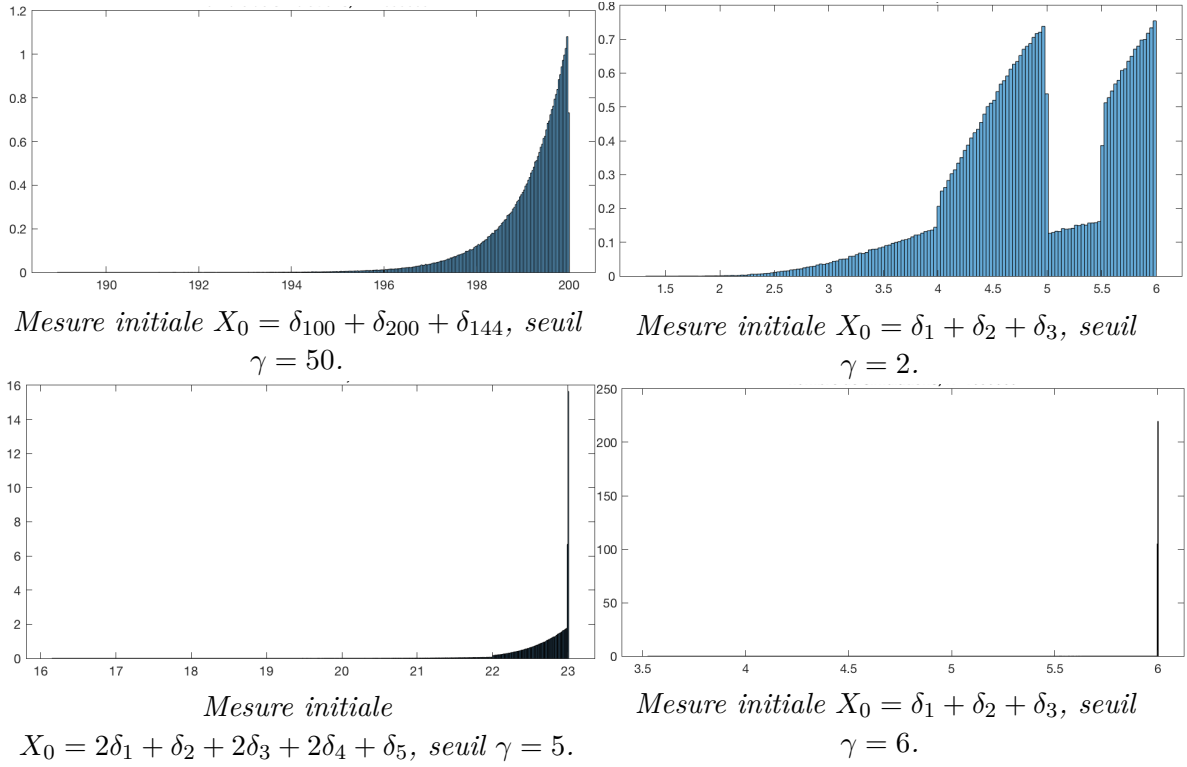
où  $\alpha > 0$  est le paramètre dans le taux de saut  $l : l(x) = x^\alpha$  pour  $x \in \mathbb{R}$ . On peut donc écrire :

$$\mathbb{E}_\mu[h(\mathbb{W}_0(Z_1))] = \sum_{i=1}^2 \int_{\mathbb{R}} h(3\gamma e^{-rt} + u_i(t)) f_{T_1}(t) \mathbb{P}(I_1 = i) dt, \quad (2.3.1)$$

avec pour  $t > 0$

$$\begin{aligned} u_1(t) &= (x_2 - \gamma e^{-rt}) \mathbf{1}_{x_2 < \gamma e^{-rt}} + (x_1 - 2\gamma e^{-rt}) \mathbf{1}_{x_1 < 2\gamma e^{-rt}}, \\ u_2(t) &= (x_1 - \gamma e^{-rt}) \mathbf{1}_{x_1 < \gamma e^{-rt}} + (x_2 - 2\gamma e^{-rt}) \mathbf{1}_{x_2 < 2\gamma e^{-rt}}. \end{aligned}$$

Afin de simplifier les calculs, on introduit les notations suivantes.



**FIGURE 2.3.1 – différents histogrammes de  $V_0(Z_1)$**

Paramètres communs : nombre de simulations =  $10^5$ , taux de croissance  $r = 2$ , puissance intensité  $\alpha = 1$

**Notation 2.3.1.**

Pour  $\gamma, r, \alpha, x_1, x_2$  dans  $\mathbb{R}_+^*$ , on note :

$$\forall i \in \{1, 2\}, \quad r_i := \frac{1}{r} \ln \left( \frac{\gamma}{x_i} \right),$$

$$r_i^{(2)} := \frac{1}{r} \ln \left( \frac{\gamma}{x_i/2} \right) = r_i + \frac{\ln 2}{r},$$

$$X^\alpha := x_1^\alpha + x_2^\alpha.$$

Alors  $\{x_i < 2\gamma e^{-rt}\} = \{t < r_i^{(2)}\}$  et  $\{x_i < \gamma e^{-rt}\} = \{t < r_i\} \subset \{t < r_i^{(2)}\}$ . On peut ainsi réécrire l'équation (2.3.1) en

$$\mathbb{E}_\mu[h(V_0(Z_1))] = \sum_{i=1}^2 \int_{\mathbb{R}} h \left( 3\gamma e^{-rt} + u_i(t) \right) x_i^\alpha e^{r\alpha t} \exp \left( -\frac{X^\alpha}{r\alpha} (e^{r\alpha t} - 1) \right) \mathbf{1}_{t>0} dt,$$

où les  $u_i$  se réexpriment en

$$u_1(t) = (x_2 - \gamma e^{-rt}) \mathbf{1}_{t < r_2} + (x_1 - 2\gamma e^{-rt}) \mathbf{1}_{t < r_1^{(2)}},$$

$$u_2(t) = (x_1 - \gamma e^{-rt}) \mathbf{1}_{t < r_1} + (x_2 - 2\gamma e^{-rt}) \mathbf{1}_{t < r_2^{(2)}}.$$



On a ainsi une expression des indicatrices seulement en fonction de  $t$ , ce qui permet de séparer les intégrales. En comparant les différents produits d'indicatrices dans  $\mathbb{V}_0$ , on remarque qu'il faut regarder les signes de  $r_i$  et  $r_i^{(2)}$ . Cela revient à étudier les relations d'ordre entre  $\gamma$ ,  $x_1/2$ ,  $x_2/2$ ,  $x_1$ ,  $x_2$ , ce qui nous donne plusieurs cas :

Cas  $\gamma \leq \min\{x_i/2\}$ , figure 2.3.2

Alors

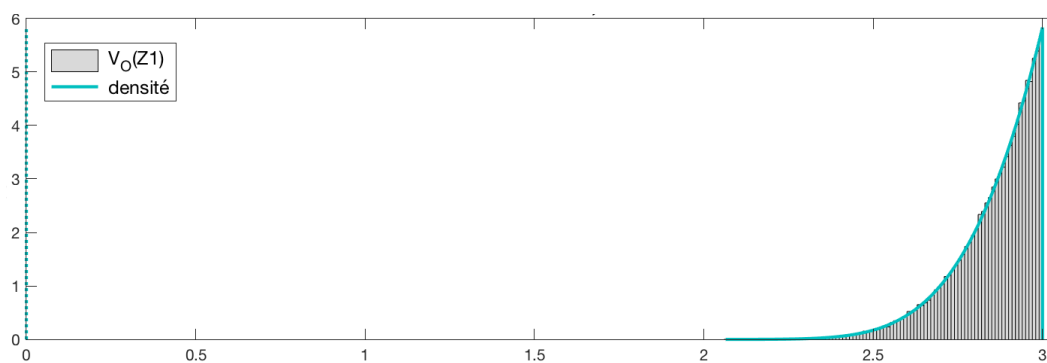
$$\begin{cases} r_1 < r_1 + \frac{\ln(2)}{r} \leq 0 \\ r_2 < r_2 + \frac{\ln(2)}{r} \leq 0 \end{cases} \implies \mathbf{E}_\mu[h(\mathbb{V}_0(Z_1))] = \int_0^\infty h(3\gamma e^{-rt}) f_{T_1}(t) dt \times 1 \\ = \int_0^{3\gamma} h(u) \frac{f_{T_1}\left(-\frac{1}{r} \ln\left(\frac{u}{3\gamma}\right)\right)}{ru} du.$$

On fait le changement de variable :

$$\begin{cases} u = 3\gamma e^{-rt} \in [3\gamma, 0], \\ t = -\frac{1}{r} \ln\left(\frac{u}{3\gamma}\right), \\ dt = -\frac{1}{ru} du, \end{cases}$$

avec  $f_{T_1}\left(-\frac{1}{r} \ln\left(\frac{u}{3\gamma}\right)\right) = X^\alpha \left(\frac{3\gamma}{u}\right)^\alpha \exp\left(-\frac{X^\alpha}{r\alpha} \left(\frac{3^\alpha \gamma^\alpha}{u^\alpha} - 1\right)\right)$ . Et après calculs, on a la densité suivante, définie sur  $\mathbb{R}$  :

$$u \mapsto \frac{3^\alpha \gamma^\alpha X^\alpha}{ru^{\alpha+1}} \exp\left(\frac{X^\alpha}{r\alpha} \left(1 - \frac{3^\alpha \gamma^\alpha}{u^\alpha}\right)\right) \mathbb{1}_{0 < u < 3\gamma}.$$



**FIGURE 2.3.2 – Histogramme et densité de  $\mathbb{V}_0(Z_1)$**   
mesure initiale  $\delta_2 + \delta_3$ , seuil  $\gamma = 1$ ,  $r = 2$ ,  $\alpha = 1$ ,  $nb\ simu = 10^5$

Cas  $x_2/2 < \gamma \leq \min\{x_2, x_1/2\}$ , figure 2.3.3

Alors on a :

$$\begin{cases} r_1 < r_1 + \frac{\ln(2)}{r} \leq 0, \\ r_2 < 0 \leq r_2 + \frac{\ln(2)}{r}, \end{cases}$$

et donc

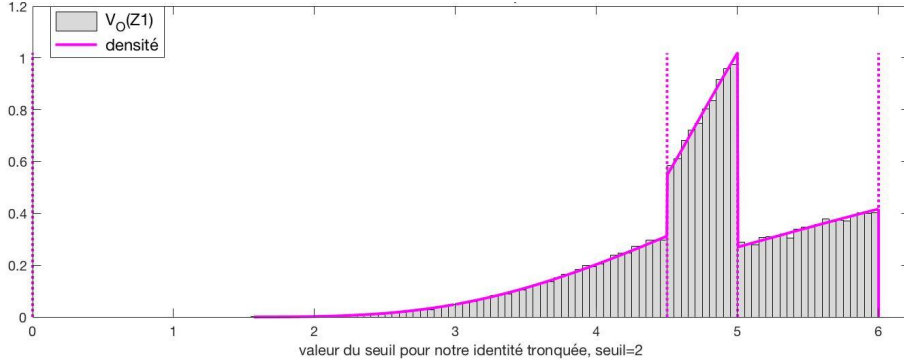
$$\mathbb{E}_\mu[h(\mathbb{V}_0(Z_1))] = \frac{x_1^\alpha}{X^\alpha} \int_0^\infty h(3\gamma e^{-rt}) f_{T_1}(t) dt + \frac{x_2^\alpha}{X^\alpha} \left[ \int_0^{r_2^{(2)}} h(\gamma e^{-rt} + x_2) f_{T_1}(t) dt + \int_{r_2^{(2)}}^\infty h(3\gamma e^{-rt}) f_{T_1}(t) dt \right].$$

Le premier terme dans la somme se traite comme dans le cas précédent ; pour le deuxième terme, on fait également deux changements de variable.

$$\begin{cases} u = \gamma e^{-rt} + x_2 \in [\gamma + x_2, \frac{3}{2}x_2], \\ t = -\frac{1}{r} \ln\left(\frac{u-x_2}{\gamma}\right), \\ dt = -\frac{1}{r(u-x_2)} du \end{cases} \quad \text{et} \quad \begin{cases} v = 3\gamma e^{-rt} \in [\frac{3}{2}x_2, 0] \\ t = -\frac{1}{r} \ln\left(\frac{v}{3\gamma}\right) \\ dt = -\frac{1}{rv} dv. \end{cases}$$

Et après calculs, on a la densité suivante :

$$u \mapsto \frac{\gamma^\alpha}{r} e^{\frac{X^\alpha}{r\alpha}} \left[ \frac{3^\alpha}{u^{\alpha+1}} \exp\left(-\frac{3^\alpha \gamma^\alpha X^\alpha}{r\alpha u^\alpha}\right) (x_1^\alpha \mathbb{1}_{0 < u < 3\gamma} + x_2^\alpha \mathbb{1}_{0 < u < \frac{3x_2}{2}}) + \frac{x_2^\alpha}{(u-x_2)^{\alpha+1}} \exp\left(-\frac{\gamma^\alpha X^\alpha}{r\alpha(u-x_2)^\alpha}\right) \mathbb{1}_{\frac{3x_2}{2} < u < x_2 + \gamma} \right].$$



**FIGURE 2.3.3 – Histogramme et densité de  $\mathbb{V}_0(Z_1)$**   
mesure initiale  $\delta_3 + \delta_5$ , seuil  $\gamma = 2$ ,  $r = 2$ ,  $\alpha = 1$ ,  $nb \text{ simu} = 10^5$

Cas  $x_2 < \gamma \leq x_1/2$ , figure 2.3.4

Alors on a :

$$\begin{cases} r_1 < r_1 + \frac{\ln(2)}{r} \leq 0, \\ 0 < r_2 < r_2 + \frac{\ln(2)}{r}, \end{cases}$$

et donc

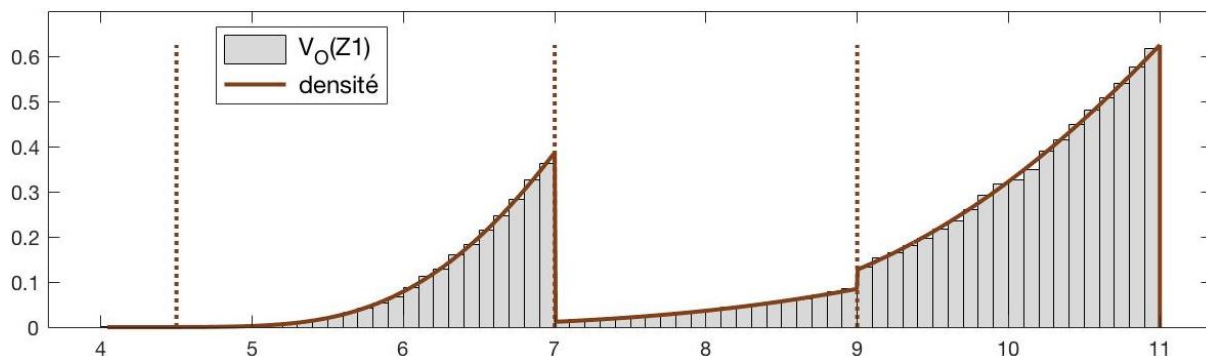
$$\mathbb{E}_\mu[h(\mathbb{V}_0(Z_1))] = \frac{x_1^\alpha}{X^\alpha} \left[ \int_0^{r_2} h(2\gamma e^{-rt} + x_2) f_{T_1}(t) dt + \int_{r_2}^\infty h(3\gamma e^{-rt}) f_{T_1}(t) dt \right] + \frac{x_2^\alpha}{X^\alpha} \left[ \int_0^{r_2^{(2)}} h(\gamma e^{-rt} + x_2) f_{T_1}(t) dt + \int_{r_2^{(2)}}^\infty h(3\gamma e^{-rt}) f_{T_1}(t) dt \right].$$

Le deuxième terme dans la somme se traite comme dans le cas précédent ; pour le premier terme, on fait également deux changements de variable.

$$\begin{cases} u = 2\gamma e^{-rt} + x_2 \in [2\gamma + x_2, 3x_2] \\ t = -\frac{1}{r} \ln\left(\frac{u-x_2}{2\gamma}\right), \\ dt = -\frac{1}{r(u-x_2)} du \end{cases} \quad \text{et} \quad \begin{cases} v = 3\gamma e^{-rt} \in [3x_2, 0] \\ t = -\frac{1}{r} \ln\left(\frac{v}{3\gamma}\right), \\ dt = -\frac{1}{rv} dv. \end{cases}$$

Et après calculs de la densité, on a la densité suivante :

$$u \mapsto \frac{\gamma^\alpha}{r} e^{\frac{X^\alpha}{r\alpha}} \left[ \frac{2^\alpha x_1^\alpha}{(u-x_2)^{\alpha+1}} \exp\left(-\frac{2^\alpha \gamma^\alpha X^\alpha}{r\alpha(u-x_2)^\alpha}\right) \mathbb{1}_{3x_2 < u < x_2+2\gamma} + \frac{3^\alpha x_1^\alpha}{u^{\alpha+1}} \exp\left(-\frac{3^\alpha \gamma^\alpha X^\alpha}{r\alpha u^\alpha}\right) \mathbb{1}_{0 < u < 3x_2} \right. \\ \left. + \frac{x_2^\alpha}{(u-x_2)^{\alpha+1}} \exp\left(-\frac{\gamma^\alpha X^\alpha}{r\alpha(u-x_2)^\alpha}\right) \mathbb{1}_{3\frac{x_2}{2} < u < x_2+\gamma} + \frac{3^\alpha x_2^\alpha}{u^{\alpha+1}} \exp\left(-\frac{3^\alpha \gamma^\alpha X^\alpha}{r\alpha u^\alpha}\right) \mathbb{1}_{0 < u < \frac{3}{2}x_2} \right].$$



**FIGURE 2.3.4 – Histogramme et densité de  $V_0(Z_1)$**   
mesure initiale  $\delta_3 + \delta_{10}$ , seuil  $\gamma = 4$ ,  $r = 2$ ,  $\alpha = 1$ , nb simu =  $10^5$

Cas  $x_2 \leq x_1/2 < \gamma < x_1$ , figure 2.3.5

Alors on a :

$$r_2^{(2)} > r_2 \geq r_1^{(2)} > 0 > r_1,$$

d'où

$$\begin{aligned} & \mathbb{E}_\mu[h(\mathbf{V}_0(Z_1))] \\ &= \frac{x_1^\alpha}{X^\alpha} \left[ \int_0^{r_1^{(2)}} h(x_1 + x_2) f_{T_1}(t) dt + \int_{r_1^{(2)}}^{r_2} h(2\gamma e^{-rt} + x_2) f_{T_1}(t) dt + \int_{r_2}^\infty h(3\gamma e^{-rt}) f_{T_1}(t) dt \right] \\ &+ \frac{x_2^\alpha}{X^\alpha} \left[ \int_0^{r_2^{(2)}} h(\gamma e^{-rt} + x_2) f_{T_1}(t) dt + \int_{r_2^{(2)}}^\infty h(3\gamma e^{-rt}) f_{T_1}(t) dt \right]. \end{aligned}$$

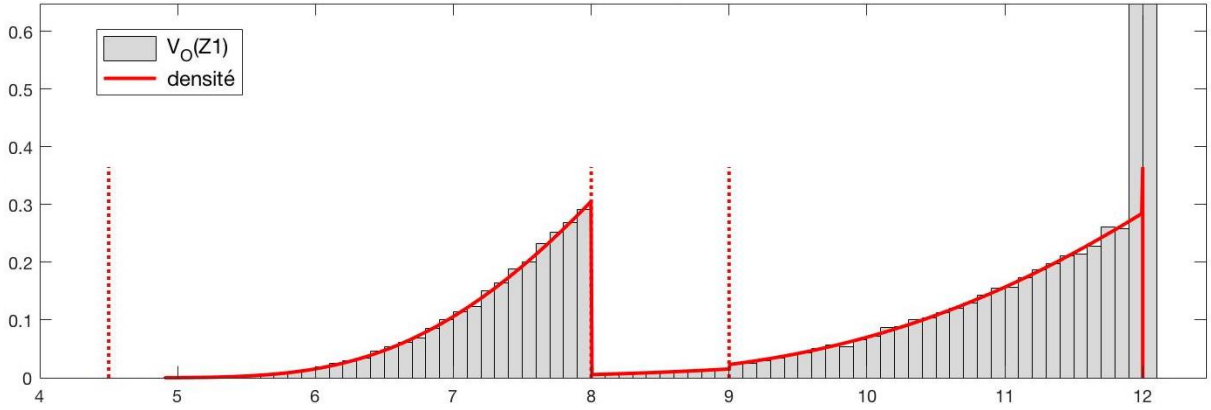
On reconnaît les changement de variables vu précédemment. Après calculs, on obtient la loi mixte suivante, somme d'une loi continue et d'une loi discrète. La partie continue est

donnée par la fonction suivante :

$$\begin{aligned}
u \mapsto & \frac{\gamma^\alpha e^{\frac{X^\alpha}{r\alpha}}}{r(u-x_2)^{\alpha+1}} \left[ 2^\alpha x_1^\alpha \exp\left(-\frac{2^\alpha \gamma^\alpha X^\alpha}{r\alpha(u-x_2)^\alpha}\right) \mathbf{1}_{3x_2 < u < x_2+x_1} \right. \\
& + x_2^\alpha \exp\left(-\frac{\gamma^\alpha X^\alpha}{r\alpha(u-x_2)^\alpha}\right) \mathbf{1}_{3\frac{x_2}{2} < u < x_2+\gamma} \left. \right] du \\
& + \frac{3^\alpha \gamma^\alpha}{ru^{\alpha+1}} e^{\frac{X^\alpha}{r\alpha}} \exp\left(-\frac{3^\alpha \gamma^\alpha X^\alpha}{r\alpha u^\alpha}\right) \left[ x_1^\alpha \mathbf{1}_{0 < u < 3x_2} + x_2^\alpha \mathbf{1}_{0 < u < \frac{3}{2}x_2} \right] du,
\end{aligned}$$

et la partie discrète est donnée par l'atome suivant :

$$\delta_{x_1+x_2}(du) \left( 1 - e^{\frac{X^\alpha}{r\alpha}} \exp\left(-\frac{2^\alpha \gamma^\alpha X^\alpha}{r\alpha x_1^\alpha}\right) \right) \frac{x_1^\alpha}{X^\alpha}.$$



**FIGURE 2.3.5 – Histogramme et densité de  $V_0(Z_1)$**   
*mesure initiale  $\delta_9 + \delta_3$ , seuil  $\gamma = 5$ ,  $r = 2$ ,  $\alpha = 1$ ,  $nb\ simu = 10^5$*

Cas  $x_2 < x_1/2 < x_1 \leq \gamma$ , figure 2.3.6

Alors on a :

$$r_2^{(2)} > r_2 > r_1^{(2)} > r_1 > 0$$

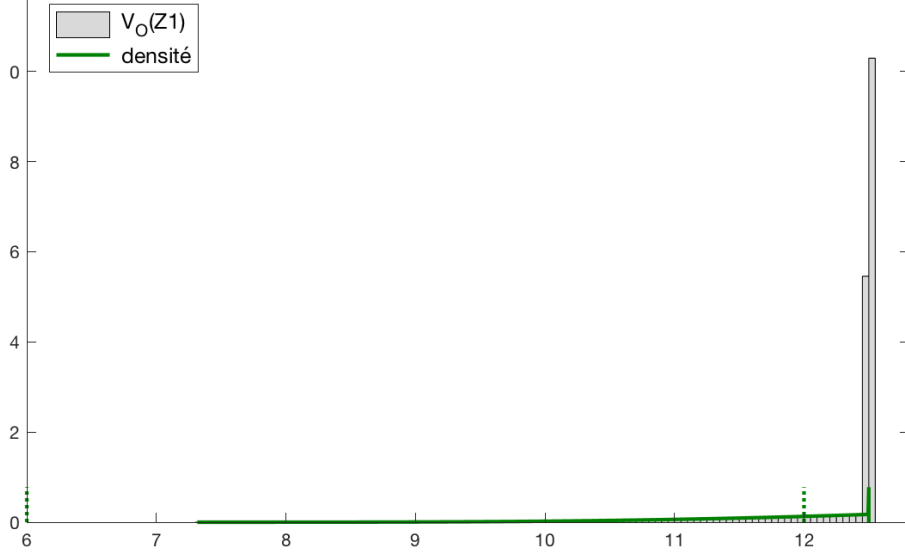
$$\begin{aligned}
& \mathbb{E}_\mu[h(\mathbf{V}_0(Z_1))] \\
& = \frac{x_1^\alpha}{X^\alpha} \left[ \int_0^{r_1^{(2)}} h(x_1+x_2) f_{T_1}(t) dt + \int_{r_1^{(2)}}^{r_2} h(2\gamma e^{-rt} + x_2) f_{T_1}(t) dt + \int_{r_2}^\infty h(3\gamma e^{-rt}) f_{T_1}(t) dt \right] \\
& + \frac{x_2^\alpha}{X^\alpha} \left[ \int_0^{r_1} h(x_1+x_2) f_{T_1}(t) dt + \int_{r_1}^{r_2^{(2)}} h(\gamma e^{-rt} + x_2) f_{T_1}(t) dt + \int_{r_2^{(2)}}^\infty h(3\gamma e^{-rt}) f_{T_1}(t) dt \right].
\end{aligned}$$

Après calculs, on obtient la loi mixte suivante, somme d'une loi continue et d'une loi discrète. La partie continue est donnée par la fonction suivante :

$$\begin{aligned}
u \mapsto & \frac{\gamma^\alpha}{r(u-x_2)^{\alpha+1}} e^{\frac{X^\alpha}{r\alpha}} \left[ 2^\alpha x_1^\alpha \exp\left(-\frac{2^\alpha \gamma^\alpha X^\alpha}{r\alpha(u-x_2)^\alpha}\right) \mathbf{1}_{3x_2 < u < x_2+x_1} \right. \\
& + x_2^\alpha \exp\left(-\frac{\gamma^\alpha X^\alpha}{r\alpha(u-x_2)^\alpha}\right) \mathbf{1}_{3\frac{x_2}{2} < u < x_2+x_1} \left. \right] du,
\end{aligned}$$

et la partie discrète est donnée par l'atome suivant :

$$\frac{1}{X^\alpha} \delta_{x_1+x_2}(\text{du}) \left[ \left( 1 - e^{\frac{X^\alpha}{r\alpha}} \exp\left(-\frac{2^\alpha \gamma^\alpha X^\alpha}{r\alpha x_1^\alpha}\right) \right) x_1^\alpha + \left( 1 - e^{\frac{X^\alpha}{r\alpha}} \exp\left(-\frac{\gamma^\alpha X^\alpha}{r\alpha x_1^\alpha}\right) \right) x_2^\alpha \right].$$



**FIGURE 2.3.6 – Histogramme et densité de  $V_0(Z_1)$**   
mesure initiale  $\delta_8 + \delta_3$ , seuil  $\gamma = 9$ ,  $r = 2$ ,  $\alpha = 1$ , nb simu =  $10^5$

Cas  $x_2/2 \leq x_1/2 < \gamma < x_2$ , figure 2.3.7

Alors on a :

$$r_2^{(2)} > r_1^{(2)} > 0 > r_2 > r_1$$

d'où :

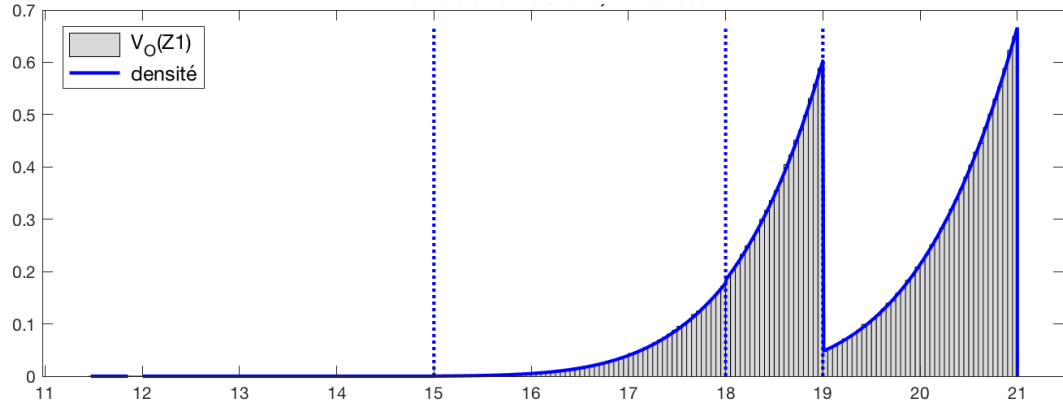
$$\begin{aligned} \mathbb{E}_\mu[h(\mathbf{V}_0(Z_1))] &= \frac{x_1^\alpha}{X^\alpha} \left[ \int_0^{r_1^{(2)}} h(\gamma e^{-rt} + x_1) f_{T_1}(t) dt + \int_{r_1^{(2)}}^\infty h(3\gamma e^{-rt}) f_{T_1}(t) dt \right] \\ &+ \frac{x_2^\alpha}{X^\alpha} \left[ \int_0^{r_2^{(2)}} h(\gamma e^{-rt} + x_2) f_{T_1}(t) dt + \int_{r_2^{(2)}}^\infty h(3\gamma e^{-rt}) f_{T_1}(t) dt \right]. \end{aligned}$$

Et après calculs, on a la densité suivante :

$$\begin{aligned} u \mapsto \frac{\gamma^\alpha}{r} e^{\frac{X^\alpha}{r\alpha}} &\left[ \frac{x_1^\alpha}{(u-x_1)^{\alpha+1}} \exp\left(-\frac{\gamma^\alpha X^\alpha}{r\alpha(u-x_1)^\alpha}\right) \mathbb{1}_{3\frac{x_1}{2} < u < x_1 + \gamma} \right. \\ &\left. + \frac{x_2^\alpha}{(u-x_2)^{\alpha+1}} \exp\left(-\frac{\gamma^\alpha X^\alpha}{r\alpha(u-x_2)^\alpha}\right) \mathbb{1}_{3\frac{x_2}{2} < u < x_2 + \gamma} \right]. \end{aligned}$$

Cas  $x_1/2 < x_2 \leq \gamma < x_1$ , figure 2.3.8

On retrouve les même types de calculs que précédemment, avec une loi mixte composée



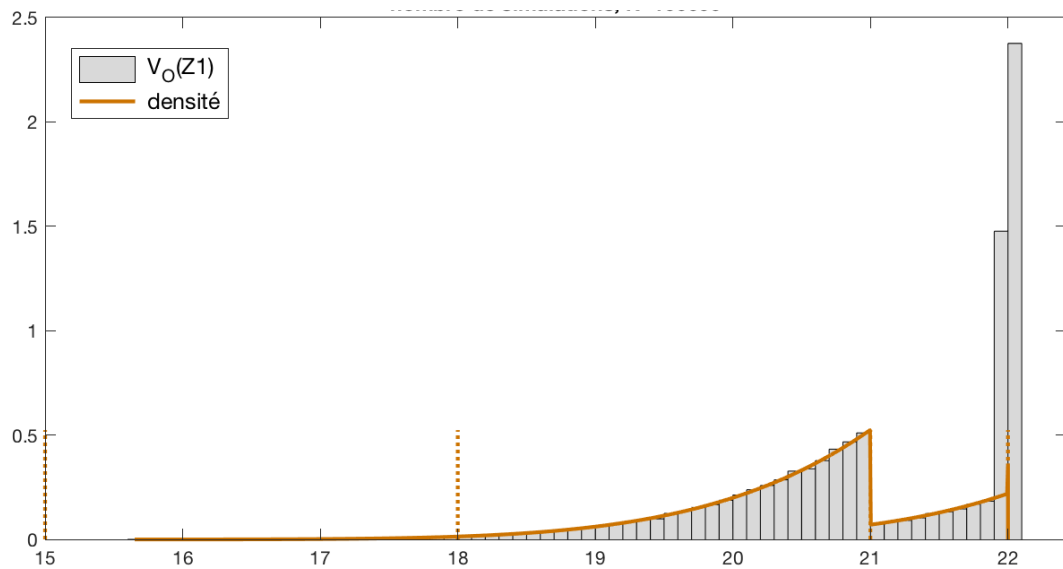
**FIGURE 2.3.7 – Histogramme et densité de  $V_0(Z_1)$**   
*mesure initiale  $\delta_{12} + \delta_{10}$ , seuil  $\gamma = 9$ ,  $r = 2$ ,  $\alpha = 1$ ,  $nb\ simu = 10^5$*

d'un partie continue :

$$u \mapsto \frac{\gamma^\alpha}{r} e^{\frac{X^\alpha}{r\alpha}} \left[ \frac{x_1^\alpha}{(u - x_1)^{\alpha+1}} \exp\left(-\frac{\gamma^\alpha X^\alpha}{r\alpha(u - x_1)^\alpha}\right) \mathbb{1}_{3\frac{x_1}{2} < u < x_1 + x_2} + \frac{x_2^\alpha}{(u - x_2)^{\alpha+1}} \exp\left(-\frac{\gamma^\alpha X^\alpha}{r\alpha(u - x_2)^\alpha}\right) \mathbb{1}_{3\frac{x_2}{2} < u < x_2 + \gamma} \right] du,$$

et d'une partie discrète

$$\frac{x_1^\alpha}{X^\alpha} \left( 1 - e^{\frac{X^\alpha}{r\alpha}} \exp\left(-\frac{\gamma^\alpha X^\alpha}{r\alpha x_2^\alpha}\right) \right) \delta_{x_1 + x_2}(du).$$



**FIGURE 2.3.8 – Histogramme et densité de  $V_0(Z_1)$**   
*mesure initiale  $\delta_{12} + \delta_{10}$ , seuil  $\gamma = 11$ ,  $r = 2$ ,  $\alpha = 1$ ,  $nb\ simu = 10^5$*

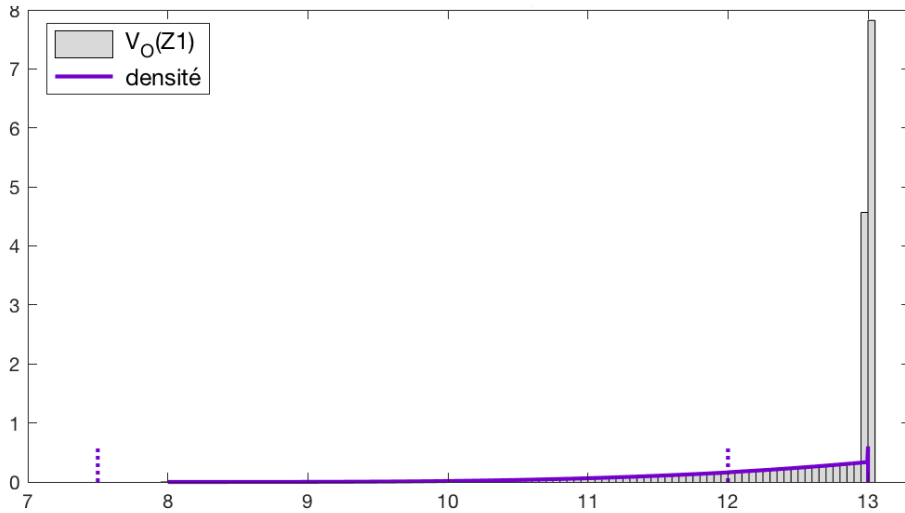
Cas  $x_2/2 < x_1/2 < x_2 \leq x_1 < \gamma$ , figure 2.3.9

On retrouve les même types de calculs que précédemment, avec une loi mixte composée d'un partie continue :

$$u \mapsto \frac{\gamma^\alpha}{r} e^{\frac{X^\alpha}{r\alpha}} \left[ \frac{x_1^\alpha}{(u-x_1)^{\alpha+1}} \exp\left(-\frac{\gamma^\alpha X^\alpha}{r\alpha(u-x_1)^\alpha}\right) \mathbb{1}_{3\frac{x_1}{2} < u < x_1+x_2} + \frac{x_2^\alpha}{(u-x_2)^{\alpha+1}} \exp\left(-\frac{\gamma^\alpha X^\alpha}{r\alpha(u-x_2)^\alpha}\right) \mathbb{1}_{3\frac{x_2}{2} < u < x_2+x_1} \right] du,$$

et d'une partie discrète :

$$\left[ \frac{x_1^\alpha}{X^\alpha} \left( 1 - e^{\frac{X^\alpha}{r\alpha}} \exp\left(-\frac{\gamma^\alpha X^\alpha}{r\alpha x_2^\alpha}\right) \right) + \frac{x_2^\alpha}{X^\alpha} \left( 1 - e^{\frac{X^\alpha}{r\alpha}} \exp\left(-\frac{\gamma^\alpha X^\alpha}{r\alpha x_1^\alpha}\right) \right) \right] \delta_{x_1+x_2}(du).$$



**FIGURE 2.3.9 – Histogramme et densité de  $V_0(Z_1)$**   
*mesure initiale  $\delta_8 + \delta_5$ , seuil  $\gamma = 8$ ,  $r = 2$ ,  $\alpha = 1$ ,  $nb\ simu = 10^5$*

Avec les différentes indicatrices, on peut voir les différents points de rupture des histogrammes, ainsi que la nature des sauts, vers le haut ou vers le bas.

## Conclusion

On a vu dans ce chapitre l'extension de la programmation dynamique pour les PDMP à valeur mesure, fortement inspirée de l'article de Gugerli [Gug86] pour les PDMP réels. Dans ce chapitre, on s'est placé avec des fonctions bornées sur  $E$  pour définir les opérateur de la programmation dynamique. Une suite logique de ce travail serait de s'intéresser à approcher numériquement les fonctions valeurs, comme par exemple dans [dSDG10], où l'on utilise une technique de quantification. Ici, notre espace de mesure étant de dimension infinie, il n'est pas possible de représenter facilement les fonctions valeurs car elles sont

définies sur un espace de mesures. Ces fonctions n'étant pas connues point par point, on les manipule via les variables aléatoires  $Z_n$  (représentant les positions post-saut). En effet, ce sont ces quantités qui apparaissent dans les équations de la programmation dynamique : les variables aléatoire  $\mathbb{W}_n(Z_{N-n})$  pour un horizon aléatoire fini de  $N$  sauts. Ces variables sont plus faciles à manipuler que des fonctions numériques, notamment d'un point de vue simulations. Cependant, reste le problème de la représentation numérique : en effet, cela reste des variables aléatoires à valeurs mesures et non des variables réelles (ou même des variables vectorielles).

Dans le cadre de ce problème de représenter numériquement des fonctions réelles sur des espaces de mesure, nous nous sommes penchés sur un problème de représentation graphique face à la zoologie de  $g(Z_1)$  dans le cadre d'un exemple très spécifique, ce qui montre une complexité même sur des cas simples. Les calculs de loi sont possibles, mais en partant de seulement deux cellules initiales et en s'arrêtant à un seul saut, on voit qu'on doit traiter 8 cas (non triviaux).

Dans le chapitre suivant, nous nous intéressons au cas particulier d'un PDMP réel simple. On cherche à faire l'approximation de ce PDMP via l'approximation du flot associé, et voir quel type d'erreur se propage en dehors de l'erreur du schéma. On choisit pour cela un flot linéaire, explicite, qui nous permet de regarder l'erreur due au décalage des sauts du processus approchant.





# CHAPITRE

## 3

# ETUDE DE CAS : TCP ET APPROXIMATION

### Plan du chapitre

<b>3.1</b>	<b>Notations</b>	<b>58</b>
<b>3.2</b>	<b>Description du modèle TCP</b>	<b>61</b>
<b>3.3</b>	<b>Approximation par schéma d'Euler</b>	<b>64</b>
<b>3.4</b>	<b>Erreur forte</b>	<b>69</b>
<b>3.5</b>	<b>Erreur faible : distance de Wasserstein</b>	<b>76</b>
3.5.1	Définition et quelques propriétés	76
3.5.2	Etude de la distance de Wasserstein pour des TCP et leurs schémas	79
3.5.3	Simuler cette erreur	83

Il est en général impossible d'obtenir une expression analytique du flot d'un PDMP quelconque. Pour approcher numériquement les solutions, on a recours à un schéma de discrétisation ce qui permet en particulier de simuler numériquement un PDMP. Par exemple, le schéma d'Euler correspond à un développement à l'ordre 1 du flot. Le but de ce chapitre est de traiter un cas particulier de PDMP réel, et de comparer à quel point le processus discrétisé par schéma d'Euler est proche du vrai processus. Pour ce faire, on choisit de prendre un exemple très simple de PDMP, afin d'avoir un flot explicite ; on pourra ainsi étudier l'erreur entre le PDMP et le processus discrétisé. Cela nous permet de voir que déjà sur un cas simple, l'erreur à calculer ne l'est pas, et on aura d'ailleurs pas d'expression explicite de l'erreur, seulement des majorations.

On utilise pour cela un modèle simplifié d'échange de données entre ordinateurs. On fait les simplifications suivantes :

- ❖ on ne considère pas de changement de mode donc on regarde uniquement l'état du système. En particulier, il n'y a qu'un seul flot  $\phi$  qui de plus est en dimension 1 ;
- ❖ les durées inter-sauts ne sont issues que de l'aléatoire : on ne considère pas de sauts aux frontières ;
- ❖ l'intensité  $l$  des sauts est constante, les durées entre les sauts sont donc déterminées par un processus de Poisson. Ce sont des variables aléatoires indépendantes et identiquement distribuées (*i.i.d.*) selon une loi exponentielle ;
- ❖ le flot  $\phi$  est explicite et linéaire :  $\phi(t, y) = y + t$ . Ceci permet d'avoir un schéma également simple.

Toutes ses conditions simplificatrices donnent accès à un modèle "jouet" mais à partir duquel on peut observer explicitement les erreurs d'approximation, et les majorer assez finement. Cela nous donne les moyen d'avoir des simulations numériques précises. De plus on s'affranchit de l'erreur de consistance car le flot choisi est linéaire, ce qui fait que l'approche par schéma d'Euler des solutions est exacte. On a ainsi accès seulement à une erreur du au décalage des sauts, mais pour laquelle on a seulement des majorations.

Le cas simple étudié dans ce chapitre est le processus modélisant un protocole d'échanges de données, le TCP<sup>1</sup>, dont on peut trouver une présentation également dans [CM16] par exemple. Après quelques rappels autour des processus de Poisson dans une première partie, on détaille la modélisation de ce processus. On explique ensuite le schéma utilisé pour approcher ce processus, puis les deux types d'erreurs qu'on considère sur ces processus. Il est présenté en annexe toutes les simulations utilisées dans ce chapitre, dans la partie A.3.

## 3.1 Notations

On introduit ici quelques notations générales qu'on utilisera tout le long de cette partie, ainsi que des notations plus spécifiques et des rappels à propos des lois exponentielles et des processus de Poisson, utilisés plus tard pour la démonstrations de nos résultats.

### Notation 3.1.1.

Soit  $\mu_1$  et  $\mu_2$  deux mesures de probabilités, respectivement sur  $(\Omega_1, \mathcal{F}_1)$  et  $(\Omega_2, \mathcal{F}_2)$  espaces mesurables. On note  $\mu_1 \otimes \mu_2$  la mesure sur l'espace produit  $(\Omega_1 \times \Omega_2, \mathcal{F}_1 \otimes \mathcal{F}_2)$  telle que

$$\forall (A_1, A_2) \in (\mathcal{F}_1, \mathcal{F}_2) \quad \mu_1 \otimes \mu_2(A_1 \times A_2) := \mu_1(A_1)\mu_2(A_2).$$

---

1. *Transmission control protocol*

**Notation 3.1.2.**

Soit  $(\Omega, \mathcal{F}, \mathbb{P})$  un espace probabilisé et  $(H, \|\cdot\|)$  un espace normé. On considère  $X$  et  $Y$  des variables aléatoires à valeur dans  $H$ . On note

- ❖  $\mathcal{P}(H)$  l'ensemble des mesures de probabilités sur  $H$ ,
- ❖  $(\mathbb{L}^p, \|\cdot\|_p)$  l'espace des variables aléatoires à valeurs réelles admettant un moment d'ordre  $p$  pour  $p$  dans  $[1, +\infty[$ . Il est muni de la norme suivante :

$$\|X\|_p = \mathbb{E}[|X|^p]^{1/p},$$

- ❖  $X \perp Y$  signifie  $X$  est indépendante de  $Y$ ,
- ❖  $I_p$  la matrice identité de l'espace des matrices réelles de taille  $p \times p$ ,
- ❖ On note  $[\cdot]$  pour désigner la partie entière supérieure des réels.

Comme on s'intéresse à des schémas numériques et des lois exponentielles, il semble logique de regarder une discrétisation de la loi exponentielle, via une loi géométrique.

**Rappel 3.1.3.**

Soit  $l > 0$ . Si  $X$  suit une loi exponentielle de paramètre  $l$ , alors :

**1** pour tout  $\theta > 0$ ,  $\left[\frac{X}{\theta}\right]$  suit une loi géométrique  $\mathcal{G}(1 - e^{-l\theta})$ ,

**2**  $\theta \left[\frac{X}{\theta}\right] \xrightarrow[\theta \rightarrow 0^+]{p.s.} X$ .

**Preuve.** On pourra regarder les exercices dans [Ouv98] (chap6) pour une référence précise.

**1** On veut montrer que cette variable aléatoire suit une loi géométrique. Soit  $k \in \mathbb{N}^*$ .

$$\mathbb{P}\left(\left[\frac{X}{\theta}\right] = k\right) = \mathbb{P}(X \in ]\theta(k-1), \theta k]) = \int_{\theta(k-1)}^{\theta k} l e^{-lx} \mathbb{1}_{x>0} dx = (e^{-l\theta})^{k-1} (1 - e^{-l\theta}),$$

ce qui correspond bien à la loi géométrique de paramètre  $1 - e^{-l\theta}$ .

**2** Pour tout  $\omega \in \Omega$  et pour tout  $\theta > 0$ , on peut écrire :

$$\left[\frac{X(\omega)}{\theta}\right] = \frac{X(\omega)}{\theta} + \varepsilon_\theta(\omega),$$

où  $\varepsilon_\theta(\omega)$  est dans l'intervalle  $[0, 1[$ . Donc

$$\left|\theta \left[\frac{X(\omega)}{\theta}\right] - X(\omega)\right| = \left|\theta \frac{X(\omega)}{\theta} + \theta \times \varepsilon_\theta(\omega) - X(\omega)\right| = \theta \times \varepsilon_\theta(\omega) \xrightarrow[\theta \rightarrow 0^+]{} 0,$$

car  $\varepsilon_\theta(\omega)$  est borné. On a donc montré la convergence presque sûre.  $\square$

On fait maintenant des rappels sur les processus de Poisson, ainsi que sur la loi exponentielle :

**Sur les processus de Poisson :** Soit  $(N_t)_{t \geq 0}$  un processus de Poisson de paramètre  $l$  strictement positif issu de 0. On note  $T_1, T_2, \dots$  les instants de sauts du processus de Poisson et on pose  $T_0 = 0$ . On rappelle ici quelques propriétés sur les processus de Poisson, qui serviront dans la suite de ce chapitre. On pourra se référer à [Ouv09](chap11,ex) pour les démonstrations.

**Rappel 3.1.4.**

Les durées inter-temps  $S_k := T_k - T_{k-1}$  sont indépendantes et identiquement distribuées selon une loi exponentielle de paramètre  $l$ .

**Rappel 3.1.5.**

Pour tout  $t > 0$ ,  $N_t$  suit une loi de Poisson de paramètre  $lt$  :  $\mathbb{P}(N_t = k) = e^{-lt} \frac{(lt)^k}{k!}$ .

**Rappel 3.1.6.**

Le processus de Poisson est à accroissement indépendants et stationnaires, ce qui signifie respectivement que<sup>2</sup> pour tout  $s, r > 0$  :

- 1  $N_{r+s} - N_r \perp N_r$  ;
- 2  $\mathcal{L}(N_{r+s} - N_r) = \mathcal{L}(N_s)$ .

**Rappel 3.1.7.**

Sachant que  $\{N_t = k\}$ , le  $k$ -uplet  $(T_1, \dots, T_k)$  suit la loi de statistique d'ordre d'un  $k$ -échantillon de loi uniforme sur  $[0, t]$ . La densité de cette loi est donnée par :

$$f(t_1, \dots, t_k) = \frac{k!}{t^k} \mathbb{1}_{\{0 < t_1 < \dots < t_k < t\}}.$$

En particulier, sachant  $\{N_t = 1\}$ ,  $T_1$  suit une loi uniforme sur  $[0, t]$ , ce qui permet de simplifier certains calculs avec conditionnement.

**Sur l'absence de mémoire :** La loi exponentielle est dite *sans mémoire* : si elle représente un temps de fonctionnement (d'une ampoule par exemple), le temps de fonctionnement qu'il reste à l'ampoule sachant qu'elle a déjà vécu un certain temps suit la même loi que la durée de vie d'une ampoule neuve : l'ampoule oublie son âge. Ceci se traduit mathématiquement de la façon suivante (on pourra se référer à [Ouv98](chap6,ex) pour démonstration).

2.  $\mathcal{L}$  désigne la loi d'une variable aléatoire, cf page 12.

**Rappel 3.1.8.**

Soit  $T$  une variable aléatoire de loi  $\mathcal{E}(l)$ . Alors on a la propriété suivante :

$$\forall s, t \geq 0 \quad \mathbb{P}(T > t + s \mid T > t) = \mathbb{P}(T > s).$$

## 3.2 Description du modèle TCP

On modélise les données sortantes d'un ordinateur relié à un réseau via un TCP comme suit : à chaque pas de temps, le débit augmente d'une unité, de manière déterministe. On a donc une évolution linéaire jusqu'à ce que le système sature, et alors le débit est divisé par deux. L'aléa se trouve dans les intervalles de temps entre chaque congestion, ce qui correspond aux durées inter-saut. Dans ce modèle, les instants de sauts ont un comportement de type processus de Poisson. Le débit suit donc un processus de Markov, déterministe par morceaux.

Soit  $Y_0$  une variable aléatoire positive qui représente la valeur initiale du processus. Dans les simulations, cette quantité sera déterministe. Soit  $(N_t)_{t \geq 0}$  un processus de Poisson de paramètre  $l > 0$  issu de 0. Dans la suite, on impose :

**Hypothèses 3.2.1.**

La variable  $Y_0$  est indépendante du processus  $(N_t)_{t \geq 0}$ .

On note  $T_1, T_2, \dots$  les instants de sauts du processus de Poisson (qui correspondent aux instants de congestion du débit) et on pose  $T_0 = 0$ . On note de plus  $S_k := T_k - T_{k-1}$  pour  $k$  dans  $\mathbb{N}^*$ .

On définit le processus  $Y = (Y_t)_{t \geq 0}$  par :

$$Y_t = \begin{cases} Y_{T_k} + t - T_k & \text{si } T_k \leq t < T_{k+1}, \\ \frac{1}{2}(Y_{T_k} + S_{k+1}) & \text{si } t = T_{k+1}. \end{cases} \quad (3.2.1)$$

On rappelle que les évènements suivant sont égaux :

$$\{T_k \leq t < T_{k+1}\} = \{N_t = k\}.$$

C'est encore égal à : "il y a eu  $k$  sauts du processus avant l'instant  $t$ ". A titre d'exemple, on donne ici les "premières" valeurs du processus TCP entre les sauts en fonction du nombre de sauts, à mettre en parallèle avec le schéma de la figure 3.2.1 :

$$Y_t = \begin{cases} Y_0 + t & \text{si } N_t = 0, \\ \frac{Y_0 + T_1}{2} + t - T_1 & \text{si } N_t = 1, \\ \frac{Y_0 + T_1}{4} + \frac{T_2 - T_1}{2} + t - T_2 & \text{si } N_t = 2, \\ \vdots & \\ \frac{Y_0 + T_1}{2^k} + \frac{T_2 - T_1}{2^{k-1}} + \dots + \frac{T_k - T_{k-1}}{2} + t - T_k & \text{si } N_t = k, \\ \vdots & \end{cases}$$

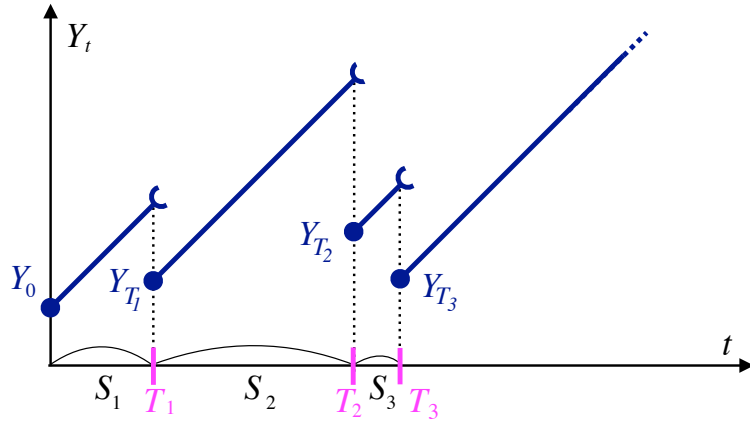


FIGURE 3.2.1 – schéma d'une trajectoire du processus TCP

Entre chaque saut, le processus  $Y$  évolue de manière déterministe par le flot  $\phi(x, t) = x + t$  où  $x$  représente la valeur au saut précédant  $t$ . A l'instant  $t$ , on se trouve entre le saut  $N_t$  et  $N_{t+1}$  et on a donc

$$Y_t = \phi(Y_{N_t}, t - T_{N_t}). \quad (3.2.2)$$

La figure 3.2.2 ci-dessus est issue d'une simulation sous MATLAB, décrite dans l'algorithme A.3.1.3.

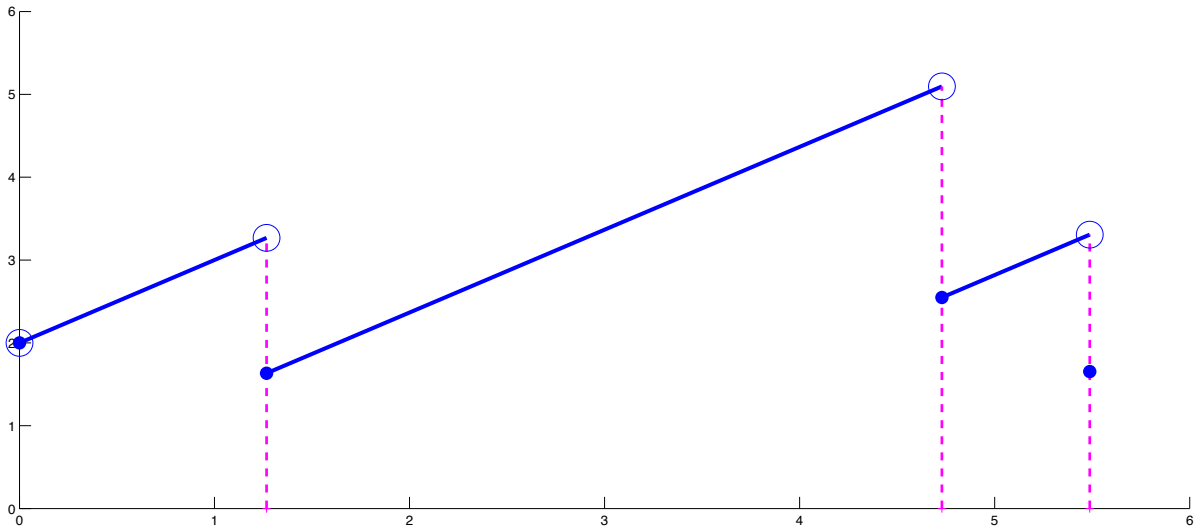


FIGURE 3.2.2 – Une trajectoire du processus TCP ( $l = 1, Y_0 = 2$ ), arrêtée au 3è saut

On définit la chaîne incluse associée à  $(Y_t)_{t \geq 0}$  comme la suite des variables des positions post-sauts, i.e.  $(Y_{T_k})_{k \in \mathbb{N}}$ , que l'on note  $(Y_n)_{n \in \mathbb{N}}$ . Les valeurs de cette chaîne sont représentées par les points pleins sur la figure 3.2.2; ce sont les limites à droite en les sauts de  $Y$ . On a la relation de récurrence suivante (par construction de  $Y$ ) :

$$Y_{k+1} = \frac{1}{2}(Y_k + S_{k+1}). \quad (3.2.3)$$

**Lemme 3.2.2.**

La chaîne incluse  $(Y_n)_{n \in \mathbb{N}}$  est une chaîne de Markov.

**Preuve.** Par la formule (3.2.3), on voit qu'on peut écrire

$$Y_{k+1} = f(Y_k, S_{k+1}),$$

où  $f$  est une fonction mesurable. Comme les  $S_i$  sont indépendants et identiquement distribués, on a directement une chaîne de Markov. □

L'aspect markovien du processus vient du fait que connaissant la valeur du processus au temps  $T_k$ , on peut déterminer la valeur du processus pour  $t \geq T_k$ . Les valeurs prises par  $Y_t$  sont conditionnellement déterministes (évolution linéaire ou division par 2), et les instants de sauts sont construits à partir d'un processus de Poisson. On formalise un peu mieux cette propriété :

**Proposition 3.2.3.**

Le processus  $(Y_t)_{t \geq 0}$  est markovien :  $\forall s, r > 0$ ,

$$\mathcal{L}(Y_{r+s} | Y_u, u \leq r) = \mathcal{L}(Y_{r+s} | Y_r). \quad (3.2.4)$$

Autrement dit, la loi du processus futur sachant tout le passé ne dépend que du présent.

**Preuve.** La démonstration dans un cas plus général est faite dans le livre de Davis [Dav93] (théorème 25.5). On a en fait une autre forme dans le cas simple du processus TCP :

On commence par écrire ce que vaut le processus en  $r + s$  à partir de  $Y_r$  :

$$\begin{aligned} Y_{r+s} &= \frac{Y_r + T_{N_{r+1}} - r}{2^{N_{r+s} - N_r}} + \frac{S_{N_{r+2}}}{2^{N_{r+s} - N_{r-1}}} + \frac{S_{N_{r+3}}}{2^{N_{r+s} - N_{r-2}}} + \dots + \frac{S_{N_{r+s}}}{2} + r + s - T_{N_{r+s}} \\ &= \frac{Y_r}{2^{N_{r+s} - N_r}} + \left\{ \frac{T_{N_{r+1}} - r}{2^{N_{r+s} - N_r}} \right\} + \left[ \frac{S_{N_{r+2}}}{2^{N_{r+s} - N_{r-1}}} + \dots + \frac{S_{N_{r+s}}}{2} \right] + \left\{ r + s - T_{N_{r+s}} \right\}. \end{aligned}$$

Cette égalité est illustrée par la figure 3.2.3. On souhaite montrer que ces termes ne dépendent du passé que par le présent, i.e. par  $Y_r$ .

Par les accroissements indépendants du processus de Poisson (rappel 3.1.6), les termes en  $N_{r+s} - N_r$  sont indépendants du passé, ce qui règle les coefficients en puissance de 2.

Les termes entre crochets ne font intervenir que des inter-temps postérieurs au présent :  $S_{N_{r+2}}, S_{N_{r+3}}, \dots, S_{N_{r+s}}$ . De plus,  $N_{r+s} = N_r + (N_{r+s} - N_r)$  où  $(N_{r+s} - N_r)$  est indépendant du passé. Comme les inter-temps sont *i.i.d.*, ils sont en particulier indépendants du passé. Reste les deux termes entre accolades, représentés par les flèches horizontales hachurées (verte à pointes vers l'extérieur et orange à pointes vers l'intérieur) sur la figure 3.2.3.



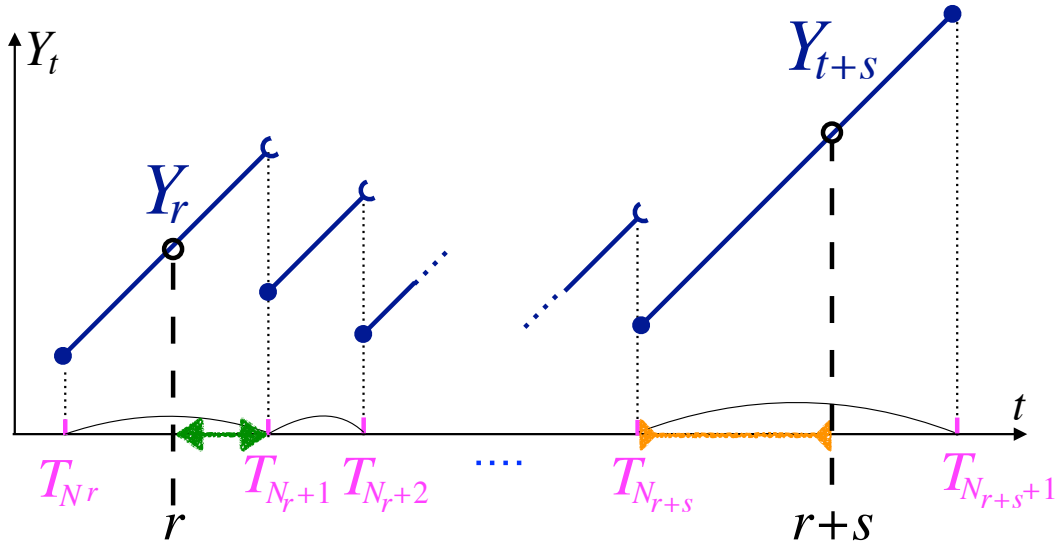


FIGURE 3.2.3 – Schéma d'une trajectoire entre les instants  $r$  et  $r + s$ .

On a :

$$\text{---} : r + s - T_{N_{r+s}} = r + s - (T_{N_{r+1}} + S_{N_{r+1}} + \dots + S_{N_{s+r}}).$$

Par le même argument que précédemment, ce terme ne dépend que du présent.

La quantité  $T_{N_{r+1}} - r$  (←→ sur le schéma) correspond au temps de vie résiduel de la variable  $S_{N_{r+1}}$  "observée" à l'instant  $r$ . On utilise la propriété d'absence de mémoire de l'exponentielle (rappel 3.1.8) : ce temps résiduel (sachant le présent) a la même loi que le premier saut d'un processus TCP partant du point  $Y_r$  (qui ne dépend donc que du présent), ce qui permet de conclure. □

Ici notre processus est simple, mais on veut l'approcher grâce à des schémas numériques, le but étant de montrer que même avec un flot linéaire explicite, on accumule quand même une erreur due à la construction de notre processus approchant.

### 3.3 Approximation par schéma d'Euler

On cherche à approcher notre processus TCP  $(Y_t)_t$  par discrétisation et schéma d'Euler, via un processus  $(Y_t^h)_{t \geq 0}$ ;  $h$  représente le pas de discrétisation choisi. Le processus approchant est construit constant par morceaux entre les pas  $0, h, 2h, 3h, \dots, nh, \dots$ , de manière à avoir un processus càdlàg (continu à droite avec limite à gauche). Il reste à choisir les valeurs de discrétisation sur les instants  $nh$ , que l'on notera  $\mathcal{Y}_n^h := Y_{nh}^h$ . On choisit ici de se placer sur une grille  $h\mathbb{N}$  pour construire notre processus approchant, mais ce n'est pas la seule possibilité. En effet, des algorithmes de simulations stochastiques, comme ceux de Gillespie [Gil76] utilisés pour de la simulation de réactions chimiques impliquant un grand nombre de molécule, intègrent dans leur grille de discrétisation les instants de sauts exacts. Comme on le verra un peu plus loin dans cette partie à la figure 3.3.2, cette considération empêche le "décalage" des processus du au retard dans le processus schéma d'Euler.

Soit  $\mathcal{Y}_0^h = Y_0^h$  une variable aléatoire positive. On suppose :

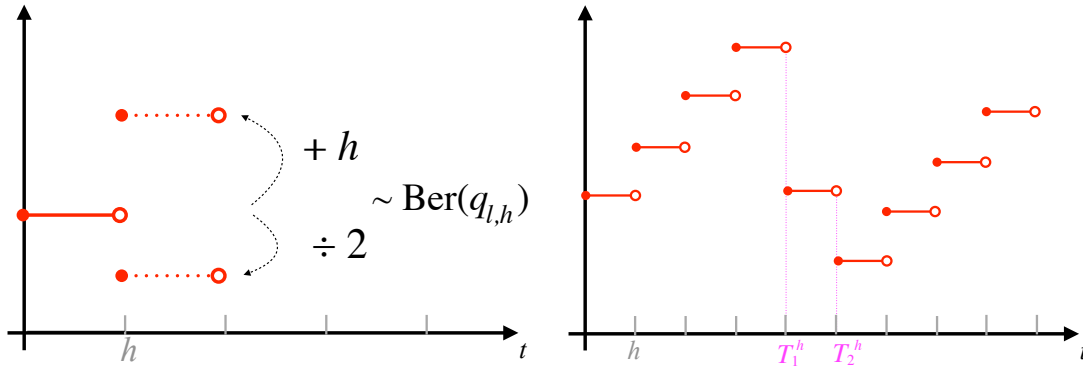
**Hypothèses 3.3.1.**

$Y_0^h$  est indépendante de tout le reste, en particulier des processus  $(N_t)_{t \geq 0}$  et  $(Y_t)_{t > 0}$ .

On rappelle qu'entre chaque saut,  $Y$  évolue de manière linéaire selon le flot  $\phi$ , par (3.2.2). On va donc se baser sur l'approximation du flot pour trouver l'approximation de notre processus. En considérant un pas de temps  $h > 0$ , on a :

$$\phi(x + h, t) = \phi(x, t) + h \times 1 + 0. \quad (3.3.1)$$

Ceci va nous donner l'approximation des morceaux déterministes. On choisira en effet une fonction constante par morceaux, qu'on incrémente de  $h$  à chaque pas de temps de longueur  $h$ . Le choix d'une grille fixe, plus simple à considérer, nous impose aussi que les instants de sauts du processus  $Y^h$  sont également sur la grille. Comme les durées inter-sauts sont strictement positives (on n'autorise pas les doubles sauts), les durées inter-sauts pour le processus  $Y^h$  seront au moins de durées  $h$ . A une position  $n \times h$  le processus aura deux possibilités pour sa valeur en  $(n + 1) \times h$ , représenté par la figure 3.3.1 : ou bien continuer la trajectoire déterministe et donc être augmenté de  $h$ , ou bien voir sa valeur divisée par deux, ce qui représente un saut.



**FIGURE 3.3.1** – illustration du schéma d'Euler : pour un pas de temps, puis jusqu'à 3 sauts.

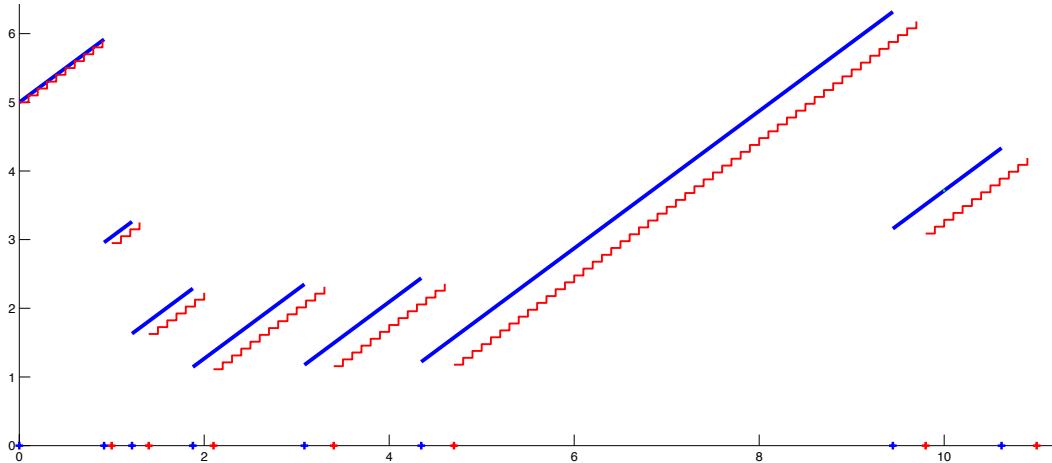
C'est donc à chaque pas  $h$  une épreuve de Bernoulli, dont on note la probabilité de succès  $q_{l,h}$ . Notre schéma d'Euler évoluant cran par cran, on en formalise l'évolution :

$$\mathcal{Y}_{n+1}^h = \begin{cases} \frac{1}{2}\mathcal{Y}_n^h & \text{avec probabilité } q, \\ \mathcal{Y}_n^h + h & \text{avec probabilité } 1 - q. \end{cases} \quad (3.3.2)$$

La dernière ligne correspond à une approximation de  $Y$  entre les sauts, approximation de la partie déterministe. Ici, l'échec dans ces épreuves de Bernoulli successives représente le saut du processus  $Y_h$ . De plus, le cran (sur la grille de discrétisation temporelle) du premier saut correspond au rang du premier succès dans une succession d'épreuves de Bernoulli indépendantes toutes de paramètre  $q$ . On note  $S_i^h$  les durées inter-sauts du

processus  $Y^h$ , et  $T_i^h$  les instants de sauts. Comme pour le processus  $Y$ , on a donc la relation  $T_n^h = \sum_{i=1}^n S_i^h$  pour tout  $n$  entier.

Les  $S_i^h$  sont à valeurs dans  $h\mathbb{N}^*$ , de même que les  $T_i^h$ . Comme presque sûrement  $S_k^h \geq h$ , le schéma d'Euler ne peut faire qu'un seul saut sur un intervalle de temps de longueur  $h$ . Le processus TCP, quant à lui, peut en faire un nombre quelconque sur une telle fenêtre; en fait, la probabilité d'avoir  $k$  sauts pendant une durée  $h$  est de  $e^{-lt} \frac{(lt)^k}{k!}$ . Ceci crée un décalage entre les deux processus, comme on peut voir sur la figure 3.3.2.



**FIGURE 3.3.2** – une trajectoire d'un processus TCP avec son approximation par schéma d'Euler ( $l = 1$ ,  $h = 0.1$ ).

Ici, on a fait une description du processus  $Y^h$  qui ne fait pas intervenir le processus de  $Y$  directement. En effet, on a donné un comportement en loi, les variables  $S_i^h$  pouvant s'écrire comme le produit de  $h$  avec une variable aléatoire de loi géométrique  $\mathcal{G}(q)$ . Mais en vu de regarder des erreurs de trajectoire illustrées par des simulations numériques, on cherche une construction faite à partir du processus initial.

Or  $Y$  saute selon des inter-temps de loi exponentielle; on cherche donc à approcher la loi exponentielle, mais à partir de loi géométrique. Or, par le rappel 3.1.3, il semble cohérent de poser pour  $k \in \mathbb{N}^*$ ,

$$S_k^h = h \times G_k^h := h \left\lceil \frac{S_k}{h} \right\rceil. \quad (3.3.3)$$

En effet, ces durées d'inter-saut pour le schéma d'Euler s'approchent de leur équivalent TCP au sens suivant :

**Lemme 3.3.2.**

Pour tout  $k \in \mathbb{N}$ , la variable  $G_k^h = \left\lceil \frac{S_k}{h} \right\rceil$  suit une loi géométrique de paramètre  $1 - e^{-lh}$ . De plus, on a la convergence suivante lorsque  $h \rightarrow 0$  :

$$\forall k \in \mathbb{N}, \quad S_k^h \xrightarrow[h \rightarrow 0]{p.s.} S_k.$$

**Preuve.** Cela résulte uniquement du rappel 3.1.3. □

Ces variables aléatoires  $S_k^h$  sont bien à valeurs dans l'ensemble  $h\mathbb{N}^*$ , car presque sûrement  $S_k > 0$  donc  $\mathbb{P}(S_k^h = 0) = 0$  et ainsi le support de  $S_k^h$  est dans  $\mathbb{R}_+^*$ . On retrouve bien les lois des variables  $G_k^h$  et  $S_k^h$  définies en (3.3.3). On en déduit également que cette chaîne  $(\mathcal{Y}_n^h)_{n \geq 0}$  est une chaîne de Markov<sup>3</sup>.

Ici, le flot est linéaire donc l'égalité (3.3.1) n'est pas une approximation, elle est exacte. Le décalage entre les deux processus n'est dû qu'à l'aléatoire et le nombre de sauts limités pour le schéma. Lorsque le flot est plus compliqué, ne serait-ce que quadratique, une erreur déterministe se rajoute, à cause de l'approximation du flot.

On a, par définition via (3.3.3),  $S_k \leq S_k^h$  pour tout  $k$ . Donc :

$$\{T_k^h \leq t\} \subset \{T_k \leq t\}.$$

On note  $(N_t^h)_{t \geq 0}$  le processus de saut associé aux instants  $T_n^h$  : pour tout  $t \geq 0$ ,  $N_t^h = \sum_{k \geq 0} \mathbb{1}_{\{T_k^h \leq t\}}$ . On a

$$N_t = \sum_{k \geq 0} \mathbb{1}_{\{T_k \leq t\}} \geq \sum_{k \geq 0} \mathbb{1}_{\{T_k^h \leq t\}} = N_t^h.$$

Heuristiquement, comme les instants de sauts  $T_k$  sont plus courts que  $T_k^h$ , le processus  $Y$  saute plus vite que  $Y^h$  donc à un instant donné il y aura plus de sauts qui se seront produits pour  $Y$  que pour  $Y^h$ .

Ce schéma d'Euler est construit à partir des inter-temps du processus TCP ; c'est un couplage particulier. On pourrait imaginer d'autres couplages, en respectant toujours la construction de la figure 3.3.1, par exemple un couplage où les deux processus sont indépendants, même si dans ce cas l'erreur en trajectoire d'un tel couplage n'aurait aucune chance de converger.

Dans la suite, on va vouloir majorer l'écart entre les deux processus, notamment en utilisant leurs moments.

**Proposition 3.3.3.**

Pour tout  $p \in \mathbb{N}$ ,  $y \geq 0$ ,  $t \geq 0$ , le moment d'ordre  $p$  de  $Y_t$  sachant  $\{Y_0 = y\}$  est donné par :

$$m_p(y, t) := \mathbb{E}_y[(Y_t)^p] = \frac{p!}{\theta_1 \dots \theta_p} + p! \sum_{m=1}^p \left( \sum_{k=0}^m \frac{y^k}{k!} \prod_{\substack{j=k \\ j \neq m}}^p \frac{1}{\theta_j - \theta_m} \right) e^{-\theta_m t}, \quad (3.3.4)$$

où  $\theta_n := l(1 - (1/2)^n)$ . De plus, quand  $t$  tend vers l'infini,  $Y_t$  converge en loi vers la loi  $\pi$  dont les moments sont donnés par la limites des moments :

$$m_p := \int x^p \pi(dx) = \frac{p!}{l^p} \frac{2^{p(p+1)/2}}{\prod_{i=1}^p (2^i - 1)}. \quad (3.3.5)$$

**Preuve.** Pour le calcul de  $m_p(y, t)$ , on renvoie au livre [CM16], dans le chapitre 17. Lorsque  $t$  tend vers l'infini, on remarque que les moments convergent tous vers des quan-

3. par le même type d'argument qu'au lemme 3.2.2

tités finies qui sont les  $m_p$ , données par :

$$m_p := \int x^p \pi(dx) = \frac{p!}{\theta_1 \dots \theta_p}.$$

En réinjectant les valeurs des  $\theta_i = l(1 - (1/2)^i)$ , on obtient bien le résultat voulu. Pour voir que ces moments sont bien les moments d'une loi, et que  $Y_t$  converge en loi vers cette même loi, on renvoie au chapitre 21 du même livre [CM16].

Cependant, on propose une preuve alternative dans la partie 3.5.1, qui utilise la distance de Wasserstein. On y montre l'existence de loi invariante et la convergence en loi du processus vers cette loi. Cette preuve est faite de manière générale également dans le chapitre 17 de [CM16], mais on la précise pour le cas du TCP à division par 2.

□

On a un résultat similaire pour le schéma d'Euler. Pour tout  $p \in \mathbb{N}$ ,  $y \geq 0$ ,  $t \geq 0$ ,  $h > 0$ , on note  $m_p^h(y, t)$  le moment d'ordre  $p$  de  $Y_t^h$  sachant  $\{Y_0^h = y\}$ . On note également  $M_p^h(y, t)$  le vecteur des moments jusqu'à l'ordre  $p$  de  $Y_t^h$  sachant  $\{Y_0^h = y\}$  :

$$\begin{aligned} M_p^h(y, t) &:= {}^t \left( \mathbb{E}_y [Y_t^h], \mathbb{E}_y [(Y_t^h)^2], \dots, \mathbb{E}_y [(Y_t^h)^p] \right) \\ &= {}^t \left( m_1^h(y, t), m_2^h(y, t), \dots, m_p^h(y, t) \right). \end{aligned}$$

**Proposition 3.3.4.**

La suite  $(\mathcal{M}_n)_{n \geq 0} := (M_p^h(y, nh))_{n \geq 0}$  vérifie la relation suivante :

$$\mathcal{M}_n = (A^{h,p})^n (\mathcal{M}_0 - (I_p - A^{h,p})^{-1} B^{h,p}) + (I_p - A^{h,p})^{-1} B^{h,p}, \quad (3.3.6)$$

où  $A^{h,p}$  est une matrice triangulaire inférieure de taille  $p \times p$  et  $B^{h,p}$  un vecteur de taille  $p$ , tou-te-s les deux ne dépendant que de  $p$ ,  $h$  et  $l$  :

$$(A^{h,p})_{i,j} = \begin{cases} 0 & \text{si } i < j \\ \frac{1+(2^i-1)e^{-lh}}{2^i} & \text{si } i = j \\ \binom{i}{j} h^{i-j} e^{-lh} & \text{si } i > j \end{cases} \quad \text{et} \quad (B^{h,p})_j = h^j e^{-lh}.$$

De plus, quand  $t$  tend vers l'infini,  $Y_t^h$  converge en loi vers la loi  $\pi^h$  dont les moments sont donnés par la limite des moments :

$$M_p^h := {}^t \left( \int x \pi_\infty^h(dx), \dots, \int x^p \pi_\infty^h(dx) \right) = (I_p - A^{h,p})^{-1} B^{h,p}.$$

**Preuve.** On a, par (3.3.2) :

$$\begin{aligned}
\mathbb{E} \left[ \left( Y_{(n+1)h}^h \right)^p \right] &= \mathbb{E} \left[ \left( Y_{nh}^h + h \right)^p \right] \mathbb{P}(0 \text{ saut au cran } n+1) + \mathbb{E} \left[ \left( \frac{Y_{nh}^h}{2} \right)^p \right] \mathbb{P}(\text{saut au cran } n+1) \\
&= \left( \sum_{j=0}^p \binom{p}{j} \mathbb{E} \left[ \left( Y_{nh}^h \right)^j \right] h^{p-j} \right) e^{-lh} + \mathbb{E} \left[ \left( Y_{nh}^h \right)^p \right] \frac{1 - e^{-lh}}{2^p} \\
&= \mathbb{E} \left[ \left( Y_{nh}^h \right)^p \right] \left( e^{-lh} + \frac{1 - e^{-lh}}{2^p} \right) + \sum_{j=1}^{p-1} \binom{p}{j} \mathbb{E} \left[ \left( Y_{nh}^h \right)^j \right] h^{p-j} e^{-lh} + h^p e^{-lh} \\
&= \mathbb{E} \left[ \left( Y_{nh}^h \right)^p \right] \left( \frac{1 + (2^p - 1)e^{-lh}}{2^p} \right) + \sum_{j=1}^{p-1} \binom{p}{j} \mathbb{E} \left[ \left( Y_{nh}^h \right)^j \right] h^{p-j} e^{-lh} + h^p e^{-lh} \\
&=: \mathbb{E} \left[ \left( Y_{nh}^h \right)^p \right] a_{p,p} + \mathbb{E} \left[ \left( Y_{nh}^h \right)^{p-1} \right] a_{p,p-1} + \dots + \mathbb{E} \left[ \left( Y_{nh}^h \right) \right] a_{p,1} + h^p e^{-lh}.
\end{aligned}$$

On en déduit :

$$\mathcal{M}_n = A^{h,p} U_{n-1} + B^{h,p} \quad \text{avec} \quad (A^{h,p})_{i,j} = \begin{cases} 0 & \text{si } i < j \\ \frac{1 + (2^i - 1)e^{-lh}}{2^i} & \text{si } i = j \\ \binom{i}{j} h^{i-j} e^{-lh} & \text{si } i > j \end{cases} \quad \text{et} \quad (B^{h,p})_j = h^j e^{-lh},$$

pour  $1 \leq i, j \leq p$ . On remarque que  $A^{h,p}$  est une matrice triangulaire inférieure, dont les coefficients diagonaux sont tous distincts ce qui assure la diagonalisabilité de  $A^{h,p}$ . Ils sont également compris strictement entre 0 et 1, ce qui assure l'inversibilité de  $I_p - A^{h,p}$ . On en déduit la formule (3.3.6), et le passage à la limite qui suit. En effet, pour tout  $n$ , la matrice  $(A^{h,p})^n$  est semblable à  $\text{diag}(a_{1,1}^n, \dots, a_{p,p}^n)$  qui converge vers la matrice nulle quand  $n$  tend vers l'infini.

De même que pour la proposition précédente, on renvoie à la partie 3.5.1 concernant la convergence en loi de  $Y_t^h$  vers la loi invariante. □

Maintenant que l'on a défini notre processus et son approximation, on cherche à quantifier l'erreur commise entre les deux. Il n'existe pas d'erreur universelle pour mesurer la précision de notre approche, mais on s'intéresse à deux types d'erreur ici. En effet, même si notre approche déterministe est exacte (le flot est linéaire), on observe quand même un décalage du processus approchant. Cette observation de trajectoires nous incite à considérer une erreur dite "forte", que l'on traite dans la partie suivante. C'est une erreur trajectorielle, donc qui dépend du couplage choisi pour les deux processus, et non pas une erreur sur leur loi.

### 3.4 Erreur forte

Soit  $T \in \mathbb{R}_+^*$  fixé. On cherche à majorer la quantité  $\mathbb{E}[|Y_t^h - Y_t|]$  pour  $t \in [0, T]$ ; plus précisément, on cherche à voir que cette quantité n'évolue pas plus vite que  $h$  :

$$\forall t \in [0, T], \quad \mathbb{E}[|Y_t^h - Y_t|] \leq C_T \times h,$$

où  $C_T$  est une constante indépendante de  $h$ . Cette erreur dépend du couplage choisi. Par exemple si on prend les deux processus indépendants, l'erreur en moment d'ordre 1 n'a aucune chance de converger.

On regarde en les instants  $t_n = nh$ , pour  $n$  dans  $\mathbb{N}$ ; comme on manipule des processus markoviens, on cherche à obtenir une relation du type

$$\mathbb{E}[|Y_h^h - Y_h|] \leq \text{Cte} \mathbb{E}[|Y_0^h - Y_0|] + \kappa(h),$$

où  $\kappa$  est une fonction. Ainsi on pourra ré-itérer au cran  $n$ . Ceci nous permet d'obtenir :

**Théorème 3.4.1.**

Pour  $n \in \mathbb{N}^*$ ,  $h > 0$ , on a :

$$\mathbb{E}[|Y_{nh}^h - Y_{nh}|] \leq \alpha_h^n \mathbb{E}[|Y_0^h - Y_0|] + \frac{1 - \alpha_h^n}{1 - \alpha_h} (\beta_h + n\gamma_h), \quad (3.4.1)$$

avec  $\alpha_h < 1$  pour tout  $h > 0$ . Les coefficients sont donnés par :

- ❖  $\alpha_h = \frac{1}{4}(1 + 3e^{-lh} + lhe^{-lh})$ ,
- ❖  $\beta_h = \frac{1}{4}l^2h^2\mathbb{E}[|Y_0^h|] + lh^2 \left(1 + \frac{3}{8}e^{-lh}\right) - \frac{5}{4l} (1 - e^{-lh} - lhe^{-lh})$ ,
- ❖  $\gamma_h = \frac{1}{4}l^2h^3$ .

**Preuve.** On commence par faire le cas  $n = 1$ . Par (3.2.1) et (3.3.2), on a différents cas possibles :

❖ Sur  $\{N_h = 0\}$  : il n'y a pas encore eu de saut donc  $T_1 > h$  et  $Y_h = Y_0 + h$ . De plus,  $N_t^h \leq N_t$  donc  $N_t^h = 0$  et de même  $Y_h^h = Y_0^h + h$ . On en déduit l'égalité suivante :

$$\begin{aligned} \mathbb{E}[|Y_h^h - Y_h| \mathbf{1}_{N_h=0}] &= \mathbb{E}[|Y_0^h + h - Y_0 - h| \mathbf{1}_{N_h=0}] \\ &= \mathbb{E}[|Y_0^h - Y_0|] e^{-lh}. \end{aligned}$$

En effet,  $N_t \sim \mathcal{P}(lt)$  (rappel 3.1.5) et  $Y_0, Y_0^h$  ont été supposées indépendantes du processus  $(N_t)_{t \geq 0}$  (par 3.2.1 et 3.3.1).

❖ Sur  $\{N_h = 1\}$  : on est dans la configuration de la figure 3.4.1. Par définition de (3.2.1), on a

$$Y_h = \frac{Y_0 + T_1}{2} + h - T_1.$$

De plus, il n'y a que deux choix pour la valeur de  $Y_h^h$  : ou bien  $N_h^h = 0$  et dans ce cas  $Y_h^h = Y_0^h + h$ ; ou bien  $N_h^h = 1$  et alors  $Y_h^h = Y_0^h/2$ .

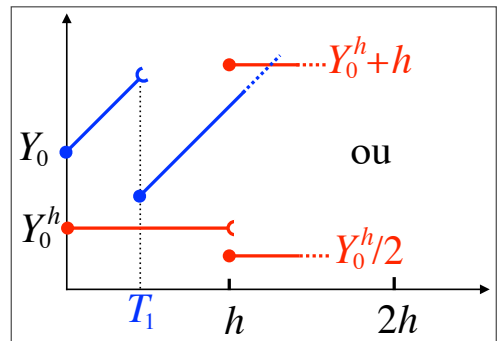


FIGURE 3.4.1 – Schéma sur  $\{N_h = 1\}$

Mais dans le premier cas :

$$\{N_h = 1, N_h^h = 0\} = \{T_1 \leq h < T_2, T_1^h > h\} \subset \left\{T_1 \leq h, h \left\lceil \frac{T_1}{h} \right\rceil > h\right\}.$$

Or :

$$\begin{aligned} \left\{T_1 \leq h, \left\lceil \frac{T_1}{h} \right\rceil > 1\right\} &= \left\{T_1 \leq h, \left\lceil \frac{T_1}{h} \right\rceil \geq 2\right\} \subset \left\{T_1 \leq h, \frac{T_1}{h} > 2 - 1 = 1\right\} \\ &= \{T_1 \leq h, T_1 > h\} \\ &= \emptyset. \end{aligned}$$

Ceci est dû à la construction  $(Y_t^h)_t$  par rapport à  $(Y_t)_t$  : en effet, s'il y a eu un saut de  $(Y_t)$  entre le pas  $mh$  et  $(m+1)h$ , alors  $Y_t^h$  a un saut en  $(m+1)h$ . Dans notre cas présent,  $m = 0$  et  $T_1 \leq h$  implique  $T_1^h = h$ . On en déduit :

$$\begin{aligned} \mathbb{E}[|Y_h^h - Y_h| \mathbf{1}_{N_h=1}] &= \mathbb{E} \left[ \left| \frac{Y_0^h}{2} - \left( \frac{Y_0 + T_1}{2} + h - T_1 \right) \right| \mathbf{1}_{N_h=1, N_h^h=1} \right] \\ &\leq \frac{1}{2} \mathbb{E}[|Y_0^h - Y_0|] \mathbb{P}(N_h = 1) + \mathbb{E}[|T_1/2 - h| \mathbf{1}_{\{N_h=1\}}] \\ &\leq \frac{1}{2} \mathbb{E}[|Y_0^h - Y_0|] e^{-lh} lh + \mathbb{E}[(h - T_1/2) \mathbf{1}_{\{N_h=1\}}]. \end{aligned}$$

Or  $\mathcal{L}(T_1 | N_t = 1) = \mathcal{U}([0; t])$  (par le rappel 3.1.7) donc :

$$\begin{aligned} \mathbb{E}[|Y_h^h - Y_h| \mathbf{1}_{N_h=1}] &= \frac{1}{2} \mathbb{E}[|Y_0^h - Y_0|] e^{-lh} lh + \mathbb{P}(N_h = 1) \times \left( h - \frac{1}{2} h \right) \\ &= l h e^{-lh} \left( \frac{1}{2} \mathbb{E}[|Y_0^h - Y_0|] + \frac{3}{4} h \right). \end{aligned}$$

❖ Sur  $\{N_h \geq 2\}$  : on a en particulier  $N_h^h = 1$  car  $T_1 < T_2 \leq h$  donc on a eu (au moins) un saut du TCP avant l'instant  $h$ .

$$\begin{aligned} \mathbb{E}[|Y_h^h - Y_h| \mathbf{1}_{N_h \geq 2}] &= \mathbb{E} \left[ \left| \frac{Y_0^h}{2} - \left( \frac{Y_0 + T_1}{4} + \frac{T_2 - T_1}{2} + h - T_2 \right) \right| \mathbf{1}_{\{N_h \geq 2\}} \right] \\ &\leq \mathbb{E} \left[ \left| \frac{Y_0^h}{4} \right| + \left| \frac{Y_0^h - Y_0}{4} \right| \right] \mathbb{P}(N_h \geq 2) + \mathbb{E} \left[ \left| \frac{3}{4} T_1 + \frac{1}{2} S_2 - h \right| \mathbf{1}_{\{N_h \geq 2\}} \right], \end{aligned}$$

car  $T_2 = T_1 + S_2$  avec  $T_1 \perp S_2$  (on a même  $(T_1, S_2) \sim \mathcal{E}(l) \otimes \mathcal{E}(l)$  par 3.1.4). On calcule l'espérance de droite à part. Comme on est sur  $\{N_h \geq 2\} = \{T_2 \leq h\}$ , on a

$$\frac{3}{4} T_1 + \frac{1}{2} S_2 \leq T_1 + S_2 = T_2 \leq h$$



donc

$$\begin{aligned}
\mathbb{E} \left[ \left| \frac{3}{4}T_1 + \frac{1}{2}S_2 - h \right| \mathbf{1}_{\{N_h \geq 2\}} \right] &= \mathbb{E} \left[ \left( h - \frac{3}{4}T_1 - \frac{1}{2}S_2 \right) \mathbf{1}_{\{T_1 + S_2 \leq h\}} \right] \\
&= \iint_{(\mathbb{R}_+^*)^2} \left( h - \frac{3}{4}x - \frac{1}{2}y \right) \mathbf{1}_{\{x+y \leq h\}} l e^{-lx} l e^{-ly} \, dx \, dy \\
&\vdots \\
&= -\frac{3}{8}lh^2 e^{-lh} + \frac{5}{4}h e^{-lh} + (1 - e^{-lh}) \left( h - \frac{5}{4l} \right) \\
&= -\frac{3}{8}lh^2 e^{-lh} + \frac{5}{4}h e^{-lh} + (1 - e^{-lh})h - (1 - e^{-lh}) \frac{5}{4l} \\
&\leq -\frac{3}{8}lh^2 e^{-lh} + \frac{5}{4}h e^{-lh} + lh^2 - (1 - e^{-lh}) \frac{5}{4l} \\
&= lh^2 \left( 1 - \frac{3}{8}e^{-lh} \right) - \frac{5}{4l} (1 - e^{-lh} - l h e^{-lh}).
\end{aligned}$$

Comme de plus  $\mathbb{P}(N_h \geq 2) = 1 - e^{-lh} - l h e^{-lh}$ , la somme de toutes ces quantités vaut :

$$\begin{aligned}
\mathbb{E}[|Y_h^h - Y_h|] &\leq \mathbb{E}[|Y_0^h - Y_0|] \times \left( e^{-lh} + \frac{lh}{2}e^{-lh} + \frac{1}{4}(1 - e^{-lh} - l h e^{-lh}) \right) \\
&\quad + \mathbb{E}[|Y_0^h|] \times \frac{1}{4}(1 - e^{-lh} - l h e^{-lh}) \\
&\quad + lh^2 \left( 1 + \frac{3}{8}e^{-lh} \right) - \frac{5}{4l} (1 - e^{-lh} - l h e^{-lh}).
\end{aligned}$$

Le premier terme est égal à  $\frac{1}{4}(1 + 3e^{-lh} + l h e^{-lh})$  et le deuxième est majoré par  $\frac{1}{4}l^2 h^2 \mathbb{E}[|Y_0^h|]$ .

On a donc une majoration en passant du cran 0 à 1. En sachant que  $\mathbb{E}[|Y_{nh}^h|] \leq \mathbb{E}[|Y_0^h|] + nh$ , ceci nous permet d'écrire la majoration suivante entre  $n$  et  $n + 1$  :

$$\begin{aligned}
\mathbb{E}[|Y_{(n+1)h} - Y_{(n+1)h}^h|] &\leq \mathbb{E}[|Y_{nh}^h - Y_{nh}|] \frac{1}{4}(1 + 3e^{-lh} + l h e^{-lh}) + \frac{1}{4}l^2 h^2 \mathbb{E}[|Y_{nh}^h|] \\
&\quad + lh^2 \left( 1 + \frac{3}{8}e^{-lh} \right) - \frac{5}{4l} (1 - e^{-lh} - l h e^{-lh}) \\
&\leq \frac{1}{4}(1 + 3e^{-lh} + l h e^{-lh}) \times \mathbb{E}[|Y_{nh}^h - Y_{nh}|] + \frac{1}{4}l^2 h^3 \times n \\
&\quad + \frac{1}{4}l^2 h^2 \mathbb{E}[|Y_0^h|] + lh^2 \left( 1 + \frac{3}{8}e^{-lh} \right) - \frac{5}{4l} (1 - e^{-lh} - l h e^{-lh}) \\
&:= \alpha_h \mathbb{E}[|Y_{nh}^h - Y_{nh}|] + \gamma_h n + \beta_h.
\end{aligned}$$

On reconnait une relation de récurrence du type  $u_{n+1} = \alpha u_n + \beta + \gamma n$ . Ceci conduit à

$$u_n = \alpha^n u_0 + \frac{1 - \alpha^n}{1 - \alpha} \beta + \gamma \sum_{k=0}^{n-1} (n - k) \alpha^k \quad \text{avec} \quad \sum_{k=0}^{n-1} (n - k) \alpha^k \leq n \sum_{k=0}^{n-1} \alpha^k = n \frac{1 - \alpha^n}{1 - \alpha}.$$

Ce qui, appliqué à l'erreur forte, nous donne l'inégalité souhaitée. □

Reste à voir le comportement des coefficients lorsque  $h$  tend vers 0, afin d'avoir une idée du comportement du schéma à l'asymptotique.

**Proposition 3.4.2.**

Lorsque  $h$  tend vers 0, on a les développements suivants :

$$\begin{aligned} \diamond \alpha_h &\underset{h \rightarrow 0}{=} 1 - \frac{lh}{2} + \frac{l^2 h^2}{8} + o(h^3), \\ \diamond \beta_h &\underset{h \rightarrow 0}{=} \left( \frac{1}{4} l^2 h^2 \mathbb{E}[|Y_0^h|] + \frac{3}{4} l \right) h^2 + \frac{1}{24} h^3 + o(h^3). \end{aligned}$$

On déduit le comportement suivant de l'erreur forte :

$$\mathbb{E}[|Y_{nh}^h - Y_{nh}|] \underset{h \rightarrow 0}{\leq} \alpha_h^n \mathbb{E}[|Y_0^h - Y_0|] + h \times [c + h \times (b + an) + o(h)] \quad (3.4.2)$$

où  $a > 0$ ,  $b > 0$  et  $c > 0$  sont des constantes qui ne dépendent que de  $l$  et  $\mathbb{E}[|Y_0^h|]$  :

$$a = \frac{l}{2}, b = \frac{l}{2} \left( \frac{1}{4} l \mathbb{E}[|Y_0^h|] + \frac{11}{12} \right) \text{ et } c = \frac{1}{2} (3 + l \mathbb{E}[|Y_0^h|])$$

De plus, les termes d'ordres supérieurs dans le développement limité de (3.4.2) (cachés dans le  $o(h)$ ) ne dépendent pas de  $n$ .

**Preuve.** Les développements des coefficients proviennent juste de développements limités.

Comme  $0 < \alpha_h < 1$ , on peut majorer directement  $\frac{1 - \alpha_h^n}{1 - \alpha_h}$  par  $\frac{1}{1 - \alpha_h}$ . Or :

$$\begin{aligned} \frac{1}{1 - \alpha_h} (\beta_h + n \gamma_h) &\underset{h \rightarrow 0}{=} \frac{2}{lh} \left( 1 + \frac{1}{4} lh + \frac{1}{16} l^2 h^2 + o(h^2) \right) \times \frac{lh^2}{4} \left( l \mathbb{E}[|Y_0^h|] + 3 + (n+1)lh + o(h) \right) \\ &\underset{h \rightarrow 0}{=} \frac{h}{2} \left[ (3 + l \mathbb{E}[|Y_0^h|]) \times h^0 + l \left( \frac{1}{4} l \mathbb{E}[|Y_0^h|] + \frac{11}{12} + n \right) \times h^1 + o(h) \right]. \end{aligned}$$

On a la majoration voulue, avec  $a = \frac{l}{2}$ ,  $b = \frac{l}{2} \left( \frac{1}{4} l \mathbb{E}[|Y_0^h|] + \frac{11}{12} \right)$  et  $c = \frac{1}{2} (3 + l \mathbb{E}[|Y_0^h|])$ .  $\square$

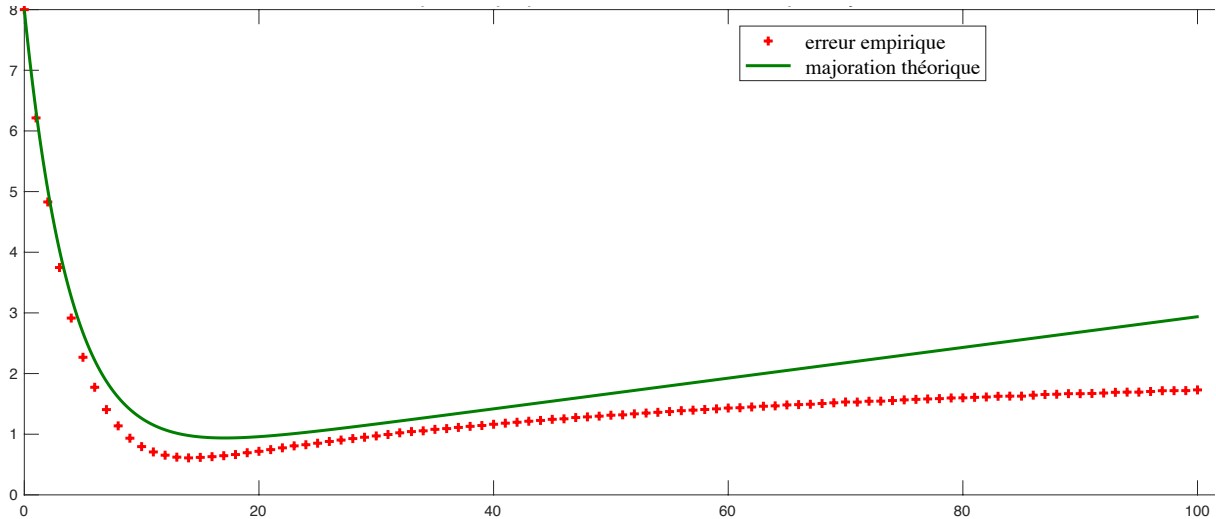
On remarque aussi que lorsque les deux processus partent du même point initial, l'erreur est majorée par un terme de l'ordre de  $h$  multiplié par une constante. En fait, lorsqu'on regarde à court terme, l'écart décroît exponentiellement selon l'erreur initiale ; puis c'est le terme affine en  $nh$  (homogène au temps) qui prend le dessus.

Cette majoration théorique (en trait continu vert sur la figure 3.4.2) tend vers l'infini lorsqu'on regarde en temps long. Ceci est dû à la partie linéaire en  $nh$ . Or, pour des  $t$  assez grands, on remarque que l'erreur empirique ne dépasse pas un certain palier, comme sur la figure 3.4.3.

A défaut d'avoir trouvé la valeur théorique de ce palier, on possède au moins deux majorations à l'infini qui permettent de borner cette erreur forte.

La première est appelée dans la figure 3.4.3 "majoration brute", c'est la courbe violette en pointillé simple. Cette majoration provient de l'inégalité triangulaire de la norme  $\|\cdot\|_1$  :

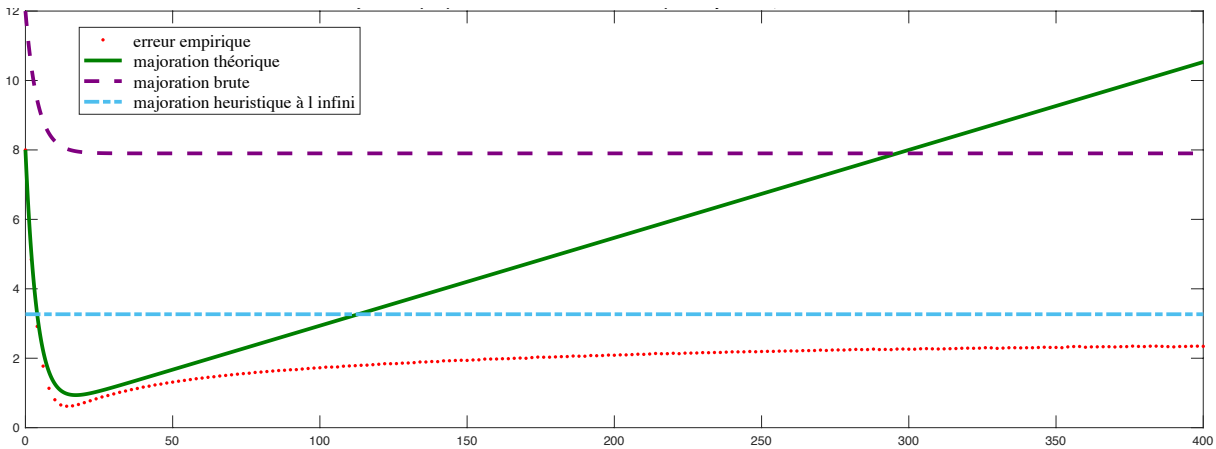
$$\forall t > 0, \quad \mathbb{E}[|Y_t - Y_t^h|] \leq \mathbb{E}[|Y_t|] + \mathbb{E}[|Y_t^h|] \xrightarrow{t \rightarrow \infty} m^1 + m^{1,h} < \infty,$$



**FIGURE 3.4.2** – *Erreur empirique ainsi que la majoration (3.4.1). L'écart est évalué avec un pas  $\Delta s = 1$ , et  $l = 0.5$ ,  $h = 0.1$ ,  $Y_0 = 10$ ,  $Y_0^h = 2$ , jusqu'à  $t = 100$ .*

par les propositions 3.3.3 et 3.3.4.

La deuxième majoration est celle appelée dans la figure 3.4.3 "majoration heuristique", c'est la courbe turquoise en pointillé double. On remarque qu'elle est plus proche de la courbe empirique que la majoration brute.



**FIGURE 3.4.3** – *Erreur empirique et diverses majorations. L'écart est évalué avec un pas  $\Delta s = 2$ , et  $l = 0.5$ ,  $h = 0.1$ ,  $Y_0 = 10$ ,  $Y_0^h = 2$ , jusqu'à  $t = 100$ .*

Cette majoration est une majoration asymptotique, valable lorsque  $t$  tend vers l'infini. Elle se base sur l'argument heuristique suivant. Au vu du couplage choisi, et de la construction du schéma d'Euler, un décalage s'opère entre les deux processus. En un temps suffisamment long, le décalage est tel qu'on peut considérer que les deux processus sont indépendants. On ne possède pas de démonstration de ce résultat, seulement une intuition. On considère les processus TCP et son schéma d'Euler en temps long, i.e. les variables aléatoires  $Y_\infty$  et  $Y_\infty^h$ . Par "décalage infini", on suppose qu'elles sont indépendantes. Ceci nous permet de déduire la majoration asymptotique suivante :

**Lemme 3.4.3.**

On a :

$$\forall h > 0, \quad \mathbb{E}[|Y_\infty - Y_\infty^h|] \leq \sqrt{\frac{16}{3l^2} + \frac{4he^{-lh}}{(1-e^{-lh})^2}} \times \left[ \frac{-2}{l} + \frac{h}{3} + e^{-lh} \left( h + \frac{2}{l} \right) \right]. \quad (3.4.3)$$

**Preuve.** Par l'inégalité de Jensen et la convexité de la fonction carrée, on a :

$$\mathbb{E}[|Y_\infty - Y_\infty^h|^2] \leq \mathbb{E}[(Y_\infty - Y_\infty^h)^2] = \mathbb{E}[(Y_\infty)^2] + \mathbb{E}[(Y_\infty^h)^2] - 2\mathbb{E}[Y_\infty]\mathbb{E}[Y_\infty^h].$$

On a utilisé l'indépendance supposée pour le dernier terme de l'égalité. On utilise maintenant les moments d'ordre 1 et 2 qu'on a déjà calculés dans les propositions 3.3.3 et 3.3.4 :

$$\begin{aligned} \mathbb{E}[Y_\infty] &= m^1 = \frac{1}{\theta_1} = \frac{1}{l(1-1/2)} = \frac{2}{l}, \\ \mathbb{E}[(Y_\infty)^2] &= m^2 = \frac{2!}{\theta_1\theta_2} = \frac{2}{l(1-1/2)l(1-1/4)} = \frac{16}{3l^2}. \end{aligned}$$

Pour les moments de  $Y_\infty^h$ , on commence par calculer la matrice  $(\text{Id}_2 - A^{h,2})$  qu'on devra ensuite inverser.

On a :

$$\begin{aligned} A^{h,2} &= \begin{pmatrix} \frac{1+e^{-lh}}{2} & 0 \\ 2he^{-lh} & \frac{1+3e^{-lh}}{4} \end{pmatrix} \implies \text{Id}_2 - A^{h,2} = \begin{pmatrix} \frac{1}{2}(1-e^{-lh}) & 0 \\ -2he^{-lh} & \frac{3}{4}(1-e^{-lh}) \end{pmatrix} \in GL_2(\mathbb{R}), \\ &\implies (\text{Id}_2 - A^{h,2})^{-1} = \frac{1}{\frac{3}{8}(1-e^{-lh})^2} \begin{pmatrix} \frac{3}{4}(1-e^{-lh}) & 0 \\ 2he^{-lh} & \frac{1}{2}(1-e^{-lh}) \end{pmatrix}, \\ &\implies (\text{Id}_2 - A^{h,2})^{-1} B^{h,2} = (\text{Id}_2 - A^{h,2})^{-1} \begin{pmatrix} he^{-lh} \\ h^2e^{-lh} \end{pmatrix}, \end{aligned}$$

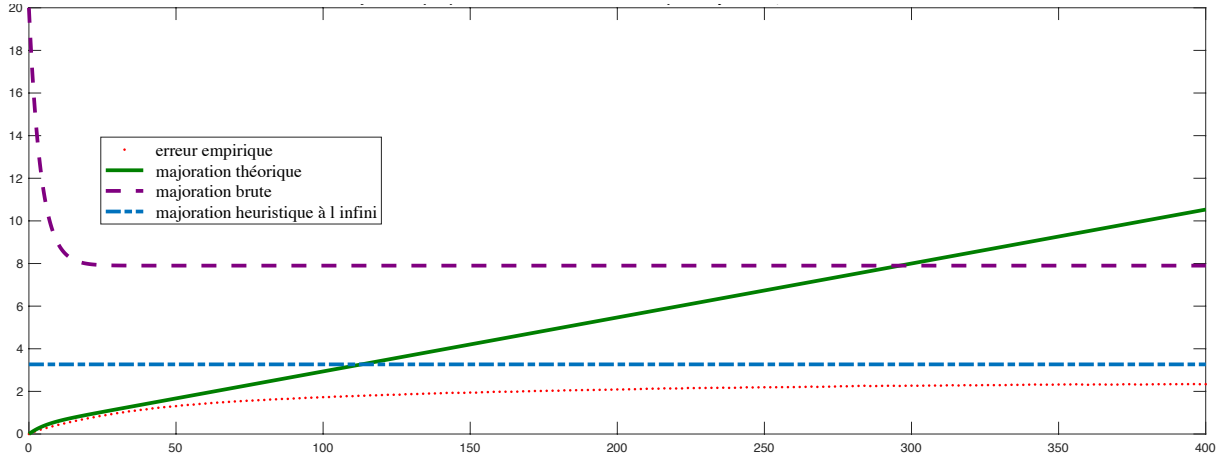
et donc

$$(\text{Id}_2 - A^{h,2})^{-1} = \begin{pmatrix} \frac{2he^{-lh}}{\frac{1-e^{-lh}}{4}} \\ \frac{4}{3} \frac{h^2e^{-lh}(3e^{-lh}+1)}{(1-e^{-lh})^2} \end{pmatrix} = \begin{pmatrix} m^{h,1} \\ m^{h,2} \end{pmatrix}.$$

On en déduit l'inégalité souhaitée en additionnant  $m^2$ ,  $m^{h,2}$  et  $(-2m^1m^{h,1})$ . □

On présente également une dernière figure 3.4.4, où l'on a fait partir les deux processus  $Y$  et  $Y^h$  du même point initial. On remarque le même type de comportement pour les différentes majorations ; la majoration théorique est la plus proche pour le court terme.

Comme on l'a vu dans cette partie, l'erreur forte dépend du couplage choisi pour la construction du schéma d'Euler. Mais est-ce le meilleur couplage possible ? On cherche une quantification de l'erreur qui ne dépend pas du couplage choisi mais qui permet quand même d'avoir une erreur précise. De plus, dans la pratique, on s'intéresse plutôt à la loi du TCP pour en déduire des informations sur son comportement (comme dans [CM16] par exemple), et donc on cherche à approcher sa loi plutôt qu'une trajectoire. C'est pour cela qu'on regarde une erreur sur les lois des processus, et non pas directement sur les trajectoires.



**FIGURE 3.4.4 – Erreur empirique et diverses majorations.** L'écart est évalué avec un pas  $\Delta s = 2$ , et  $l = 0.5$ ,  $h = 0.1$ ,  $Y_0 = 10$ ,  $Y_0^h = 10$ , jusqu'à  $t = 100$ .

### 3.5 Erreur faible : distance de Wasserstein

Précédemment, on cherchait à majorer le moment de l'erreur entre le schéma et le TCP ; on voudrait maintenant majorer l'écart de loi. Pour cela, on utilise la distance de Wasserstein, qui est une distance sur les lois.

#### 3.5.1 Définition et quelques propriétés

Soit  $p \geq 1$  et  $(H, |\cdot|)$  un espace normé. On considère les lois  $\mu$  sur  $H$  admettant un moment d'ordre  $p$  fini. On note  $\mathcal{P}(H, p)$  l'ensemble de ces lois :

$$\begin{aligned} \mathcal{P}(H, p) &:= \{ \mu \in \mathcal{P}(H) : \forall x \in H, \int_H |x - y|^p \mu(dy) < \infty \} \\ &= \{ \mu \in \mathcal{P}(H) : \exists x \in H, \int_H |x - y|^p \mu(dy) < \infty \} \end{aligned}$$

#### Définition 3.5.1.

Soit  $\mu_1$  et  $\mu_2$  dans  $\mathcal{P}(H, p)$ . La distance de Wasserstein d'ordre  $p$  entre  $\mu_1$  et  $\mu_2$  est définie par :

$$\mathcal{W}_p(\mu_1, \mu_2) = \inf_{\substack{(X, Y) \\ X \sim \mu_1 \\ Y \sim \mu_2}} \mathbb{E}[|X - Y|^p]^{1/p} = \inf_{\pi \in \Pi(\mu_1, \mu_2)} \left[ \int_{H \times H} |x - y|^p \pi(dx, dy) \right]^{1/p}.$$

où  $\Pi(\mu_1, \mu_2)$  est l'ensemble des lois sur  $H \times H$  qui ont pour marginales  $\mu_1$  et  $\mu_2$ . Si  $p = 1$ , on note  $\mathcal{W} = \mathcal{W}_1$ .

Pour calculer la distance de Wasserstein entre les lois de deux variables aléatoires  $X$  et  $Y$ , il faut donc trouver la loi jointe pour  $(X, Y)$  qui minimise la distance  $\|X - Y\|_p$  avec les lois marginales fixées. On peut penser par exemple à regarder la loi  $\mu_1 \otimes \mu_2$ , qui

revient à choisir  $X$  et  $Y$  indépendants. Dans la suite, on note par abus  $\mathcal{W}(X, Y)$  au lieu de  $\mathcal{W}(\mathcal{L}(X), \mathcal{L}(Y))$ , pour  $X, Y$  des variables aléatoires intégrables.

**Exemple 3.5.2.** Soit  $x, y$  deux réels. On considère  $\delta_x$  et  $\delta_y$  les masses de Dirac associées. Alors :

$$\mathcal{W}(\delta_x, \delta_y) = \inf_{\substack{(X, Y) \\ X=x \\ Y=y}} \mathbb{E}[|X - Y|] = |x - y|.$$

Lorsqu'on se place sur  $\mathbb{R}$ , il existe plusieurs formulations plus simples de la distance  $\mathcal{W}$ , notamment une expression explicite :

**Proposition 3.5.3.**

Soit  $\mu_1$  et  $\mu_2$  dans  $\mathcal{P}(\mathbb{R})$ , dont les fonctions de répartition respectives sont notées  $F_1$  et  $F_2$ . Alors :

$$\mathcal{W}(\mu_1, \mu_2) = \int_{\mathbb{R}} |F_1(x) - F_2(x)| dx.$$

La démonstration de cette proposition est faite, par exemple, dans le livre [Vil08], au chapitre 6.

Cette proposition permet d'exhiber un exemple plus complexe que le précédent.

**Exemple 3.5.4.** Soit  $X \sim \mathcal{E}(l)$  et  $G \sim \mathcal{G}(p)$  avec  $l > 0$  et  $p \in ]0, 1 - e^{-lh}]$ . On pose  $Y := hG$  pour  $h > 0$ . Alors

$$\mathcal{W}(X, Y) = \frac{h}{p} - \frac{1}{l}.$$

On commence par calculer  $F_Y$  : la variable  $Y$  prend ses valeurs dans l'ensemble  $h\mathbb{N}^*$ . Donc

$$\forall k \in \mathbb{N}, \quad \mathbb{P}(Y = kh) = \mathbb{P}(G = k) \implies \forall k \in \mathbb{N}, \quad F_Y(kh) = F_G(k)$$

De plus, par la proposition 3.5.3, on a

$$\begin{aligned} \mathcal{W}(X, Y) &= \int_{\mathbb{R}} \left| (1 - e^{-lx}) \mathbf{1}_{x>0} - F_Y(x) \right| dx \\ &= \sum_{k \geq 0} \int_{kh}^{(k+1)h} \left| 1 - e^{-lx} - F_Y(kh) \right| dx \\ &= \int_0^h (1 - e^{-lx}) dx + \sum_{k \geq 1} \int_{kh}^{(k+1)h} \left| 1 - e^{-lx} - (1 - (1-p)^k) \right| dx \\ &= h + \frac{e^{-lh} - 1}{l} + \sum_{k \geq 1} \int_{kh}^{(k+1)h} \left| (1-p)^k - e^{-lx} \right| dx. \end{aligned} \quad (3.5.1)$$

Dans la suite, on note  $q := 1 - p$ . On cherche pour quelles valeurs de  $x$  la quantité dans la valeur absolue est positive :

$$q^k - e^{-lx} \geq 0 \iff k \ln(q) \geq -lx \iff \frac{k \ln(1/q)}{l} \leq x. \quad (3.5.2)$$

Or, comme  $p \leq 1 - e^{-lh}$ , on a

$$q \geq e^{-lh} \implies \ln(1/q) \leq lh \implies \forall k \geq 1, \frac{1}{l} k \ln(1/q) \leq kh.$$

Mais comme  $x \in [kh, (k+1)h]$ , la suite d'équivalence (3.5.2) est vraie sur tous les intervalles d'intégration. On déduit :

$$\begin{aligned} \sum_{k \geq 1} \int_{kh}^{(k+1)h} (q^k - e^{-lx}) dx &= \sum_{k \geq 1} hq^k + \frac{1}{l} e^{-lhk} (e^{-lh} - 1) \\ &= hq \frac{q}{1-q} + \frac{e^{-lh} - 1}{l} e^{-lh} \frac{1}{1 - e^{-lh}} \\ &= h \frac{q}{p} - \frac{e^{-lh}}{l}. \end{aligned}$$

Ceci, combiné avec le résultat (3.5.1), permet de déduire la distance  $\mathcal{W}(X, Y)$ . Terminons cet exemple avec le cas particulier de  $p = 1 - e^{-lh}$ . On a alors

$$\mathcal{W}(X, Y) = \frac{h}{1 - e^{-lh}} - \frac{1}{l} \xrightarrow{h \rightarrow 0} 0.$$

□

Au vu de la définition, on a directement  $\mathcal{W}(Y_{nh}, Y_{nh}^h) \leq \mathbb{E}[|Y_{nh}^h - Y_{nh}|]$ ; en effet, la construction de notre schéma d'Euler à partir du TCP est un couplage particulier pour la loi jointe de  $(Y_t, Y_t^h)$  (à  $t$  fixé). On aurait pu par exemple construire le processus  $(Y_t^h)_t$  indépendamment du TCP, en considérant juste des lois de Bernoulli comme dans la formule (3.3.2).

Par rapport à l'erreur forte, une des différences importantes est que l'erreur de Wasserstein ne dépend pas du couplage choisi. Ceci est valable dans les calculs mais également pour les simulations, comme nous le verrons dans la partie suivante 3.5.2.

On énonce ici une propriété sur ces distances qui permet de montrer l'existence des lois limites des processus  $(Y_t)_{t \geq 0}$  et  $(Y_t^h)_{t \geq 0}$ .

### Proposition 3.5.5.

La convergence pour la distance  $\mathcal{W}_p$  est équivalente à la convergence en loi et la convergence des moments jusqu'à l'ordre  $p$  :  
soit  $(\mu_n)_n \subset \mathcal{P}(H, p)$  et  $\mu \in \mathcal{P}(H, p)$ , alors

$$\mu_n \xrightarrow[n \rightarrow \infty]{\mathcal{W}_p} \mu \iff \begin{cases} \mu_n \xrightarrow[n \rightarrow \infty]{\mathcal{L}} \mu \\ \forall k \leq p, \exists x_0 \in H, \int_H |x - x_0|^k \mu_n(dx) \xrightarrow[n \rightarrow \infty]{} \int_H |x - x_0|^k \mu(dx). \end{cases}$$

On pourra se référer à [Vil09] pour cette équivalence. De plus, cette distance permet d'avoir des bonnes propriétés pour l'espace  $\mathcal{P}(H, p)$ .

### Proposition 3.5.6.

Pour tout  $p \geq 1$ , l'espace  $(\mathcal{P}(H, p), \mathcal{W}_p)$  est un espace métrique complet.

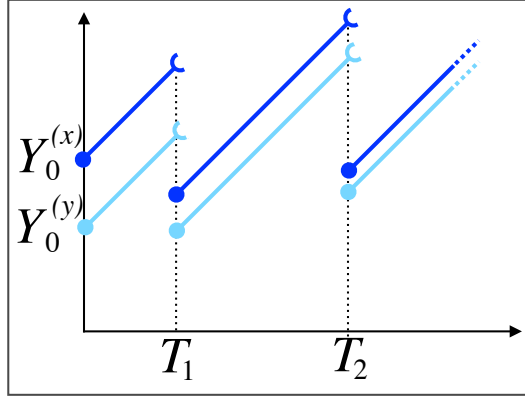


FIGURE 3.5.1 – Schéma des sauts simultanés

La démonstration de cette propriété est l’objet du théorème 5.4 dans [Che92]. Elle permet d’utiliser le théorème de point fixe pour les fonctions contractantes, et ainsi de démontrer l’existence des lois limites. Pour cela, on doit faire apparaître une contraction.

### 3.5.2 Etude de la distance de Wasserstein pour des TCP et leurs schémas

On commence cette partie par des résultats sur les distances de Wasserstein de processus TCP partant de lois initiales différentes.

#### Proposition 3.5.7.

Soit  $(Y_t^{(\mu)})_{t \geq 0}$  le processus TCP ayant pour loi initiale  $Y_0 \sim \mu$ . Soit  $\mu, \nu$  deux lois quelconques sur  $\mathbb{R}_+$  ayant un moment d’ordre  $p$ . Alors pour tout  $t \geq 0$ ,

$$\mathcal{W}_p(Y_t^{(\mu)}, Y_t^{(\nu)}) \leq \mathcal{W}_p(\mu, \nu) e^{-\frac{\theta_p}{p} t}. \quad (3.5.3)$$

**Preuve.** On fait la démonstration pour  $p = 1$ , le cas général se faisant de la même manière. Soit  $x$  et  $y$  deux positions initiales. On note  $Y_t^{(x)}$  pour désigner le TCP ayant pour valeur initiale  $Y_0 = x$  et de même pour  $Y_t^{(y)}$ . Comme on s’intéresse à des distances en loi, on peut considérer deux processus  $(Y_t^{(x)})_{t \geq 0}$  et  $(Y_t^{(y)})_{t \geq 0}$  qui utilisent le même processus de Poisson  $(N_t)_{t \geq 0}$  pour leurs instants de sauts ; cela ne change pas leur lois respectives.

On a :

$$\begin{aligned} \mathbb{E}[|Y_t^{(x)} - Y_t^{(y)}|] &= \sum_{k \geq 0} \mathbb{E}[|Y_t^{(x)} - Y_t^{(y)}| \mathbf{1}_{N_t=k}] \\ &= \sum_{k \geq 0} \mathbb{E} \left[ \frac{|x - y|}{2^k} \mathbf{1}_{N_t=k} \right]. \end{aligned}$$

En effet, l’écart entre les trajectoires reste constant entre les sauts, puis il est divisé par deux à chaque saut, comme illustré sur la figure 3.5.1.



De plus,

$$\mathbb{E}[|Y_t^{(x)} - Y_t^{(y)}|] = |x - y| \sum_{k \geq 0} 2^{-k} \frac{(lt)^k}{k!} e^{-lt} = |x - y| e^{-\frac{l}{2}t}.$$

Maintenant, on ne considère plus  $Y_0^{(x)} = x$  et  $Y_0^{(y)} = y$ . On regarde les lois  $\mu$  et  $\nu$  comme lois initiales, i.e.  $Y_0^{(\mu)} \sim \mu$  et  $Y_0^{(\nu)} \sim \nu$ . On considère un couplage quelconque  $\pi$  de  $\mu$  et  $\nu$ . Or

$$\mathcal{W}(Y_t^{(\mu)}, Y_t^{(\nu)}) \leq \mathbb{E}[|Y_t^{(\mu)} - Y_t^{(\nu)}|] = e^{-\frac{l}{2}t} \mathbb{E}[|Y_0^{(\mu)} - Y_0^{(\nu)}|] = e^{-\frac{l}{2}t} \int |x - y| \pi(dx, dy).$$

Comme on a choisi  $\pi$  quelconque, on peut prendre l'infimum sur les couplages, d'où le résultat.  $\square$

Grâce à cette contraction, on en déduit la propriété suivante :

**Théorème 3.5.8.**

Le processus  $(Y_t)_{t \geq 0}$  admet une loi invariante  $\pi$ . De plus pour toute loi initiale  $\mu$  ayant un moment d'ordre 1, le processus converge en loi et sa limite suit la loi invariante :

$$Y_t^{(\mu)} \xrightarrow[t \rightarrow \infty]{\mathcal{L}} Y_\infty \quad \text{et} \quad Y_\infty \sim \pi.$$

**Preuve.** En reprenant la propriété 3.5.7 et comme  $e^{-\frac{l}{2}t} < 1$  pour tout  $t \geq 0$  fixé, on utilise le théorème de point fixe. Donc il existe une unique mesure de probabilité  $\pi_t$  qui est le point fixe de la fonction

$$\mathfrak{F}_t : \mu \in \mathcal{P}(\mathbb{R}_+) \mapsto \mathcal{L}(Y_t^{(\mu)}).$$

Le fait que  $\pi_t = \pi_s$  pour tout  $s, t \geq 0$  utilise le fait que le processus  $Y$  est un processus de Markov. La propriété se traduit de la manière suivante pour les fonctions  $\mathfrak{F}_t$  :

$$\mathfrak{F}_{t+s} = \mathfrak{F}_{s+t} = \mathfrak{F}_t \circ \mathfrak{F}_s = \mathfrak{F}_s \circ \mathfrak{F}_t.$$

Ceci implique

$$\mathfrak{F}_{t+s}(\pi_t) = \begin{cases} \mathfrak{F}_s(\mathfrak{F}_t(\pi_t)) = \mathfrak{F}_s(\pi_t) \\ \mathfrak{F}_t(\mathfrak{F}_s(\pi_t)) \end{cases},$$

donc  $\mathfrak{F}_s(\pi_t)$  est un point fixe de  $\mathfrak{F}_t$ . Mais par unicité du point fixe, cela implique

$$\mathfrak{F}_s(\pi_t) = \pi_t,$$

et finalement la loi invariante existe :  $\pi_s = \pi_t = \pi \quad \forall s, t > 0$ . Si on prend  $\mu = \pi$  dans (3.5.3), alors

$$\mathcal{W}(Y_t^{(\pi)}, Y_t^{(\nu)}) = \mathcal{W}(\pi, \mathcal{L}(Y_t^{(\nu)})) \leq \mathcal{W}(\pi, \nu) e^{-\frac{l}{2}t} \xrightarrow[t \rightarrow \infty]{} 0.$$

On a donc convergence en distance de Wasserstein du processus TCP vers la loi invariante. Ainsi, écrire  $Y_\infty$  prend sens, et la loi de cette variable aléatoire est la loi invariante. De plus, par la propriété 3.5.5, on en déduit la convergence en loi.

□

En mimant la même démonstration, on obtient le même type de résultats pour le schéma d'Euler :

**Théorème 3.5.9.**

On a :

$$\mathcal{W}(Y_{nh}^{h,(\mu)}, Y_{nh}^{h,(\nu)}) \leq \mathcal{W}(\mu, \nu) \left( \frac{1 + e^{-lh}}{2} \right)^n. \quad (3.5.4)$$

On en déduit que la chaîne de discrétisation  $(Y_{nh}^h)_{n \geq 0}$  admet une loi invariante  $\pi^h$ . De plus, pour toute loi initiale ayant un moment d'ordre 1  $\mu$ , le processus converge en loi et sa limite est la loi invariante :

$$Y_{nh}^{h,(\mu)} \xrightarrow[n \rightarrow \infty]{\mathcal{L}} Y_\infty^h \quad \text{et} \quad Y_\infty^h \sim \pi^h.$$

**Preuve.** Avec les mêmes notations que dans la preuve précédente, on voit que :

$$\begin{aligned} \mathbb{E} \left[ \left| Y_{nh}^{h,(x)} - Y_{nh}^{h,(y)} \right| \right] &= \mathbb{E} \left[ \frac{\left| Y_{(n-1)h}^{h,(x)} - Y_{(n-1)h}^{h,(y)} \right|}{2} \right] \mathbb{P}(\text{saut au cran } n) \\ &\quad + \mathbb{E} \left[ \left| Y_{(n-1)h}^{h,(x)} - Y_{(n-1)h}^{h,(y)} \right| \right] \mathbb{P}(\text{pas de saut}) \\ &= \mathbb{E} \left[ \left| Y_{(n-1)h}^{h,(x)} - Y_{(n-1)h}^{h,(y)} \right| \right] \left( \frac{1 - e^{-lh}}{2} + e^{-lh} \right) \\ &\quad \vdots \\ &= |x - y| \left( \frac{1 + e^{-lh}}{2} \right)^n, \end{aligned}$$

d'où le premier résultat.

Comme précédemment,  $\left( \frac{1+e^{-lh}}{2} \right) < 1$  donc la fonction

$$\mathfrak{F}^h : \mu \in \mathcal{P}(\mathbb{R}_+) \mapsto \mathcal{L}(Y_h^{(\mu)})$$

est contractante pour la distance  $\mathcal{W}$ . Elle admet une unique mesure de probabilité  $\pi^h$  comme point fixe. Or, la chaîne de discrétisation est une chaîne de Markov par la construction (3.3.2), ce qui implique que  $\pi^h$  est sa mesure invariante. De plus, la variable  $Y_{(n+1)h}^h$  sachant  $\{Y_0^h \sim \mu\}$  a la même loi que la variable  $Y_h^h$  sachant  $\{Y_0^h \sim Y_{nh}^{h,(\mu)}\}$  :

$$\mathcal{L}(Y_{(n+1)h}^h \mid Y_0^h \sim \mu) = \mathcal{L}(Y_h^h \mid Y_0^h \sim Y_{nh}^{h,(\mu)}).$$

Ceci se traduit encore par :

$$\mathcal{L}(Y_{(n+1)h}^{h,(\mu)}) = \mathfrak{F}^h(\mathcal{L}(Y_{nh}^{h,(\mu)})).$$

C'est une relation du type  $u_{n+1} = f(u_n)$  pour une fonction contractante, ce qui implique que la suite  $(Y_{nh}^h)_{n \geq 0}$  converge en distance  $\mathcal{W}$  vers la loi invariante, d'où le deuxième résultat.

□

**Remarque 3.5.10.** On a montré l'existence des lois limites, mais on y a seulement accès par leur moments.

**Remarque 3.5.11.** Dans les majorations (3.5.3) et (3.5.4), on a accès à des vitesses de convergence vers les lois limites, qu'on note  $V_{TCP}$  et  $V_{Euler}$ . Pour  $t = nh$  quelconque,

$$\begin{aligned}\mathcal{W}(Y_{nh}, Y_\infty) &\leq \mathcal{W}(Y_0, Y_\infty) e^{-\frac{1}{2}nh} = \mathcal{W}(Y_0, Y_\infty) \left(e^{-\frac{1}{2}h}\right)^n =: \mathcal{W}(Y_0, Y_\infty) \times (V_{TCP})^n, \\ \mathcal{W}(Y_{nh}^h, Y_\infty^h) &\leq \mathcal{W}(Y_0^h, Y_\infty^h) \left(\frac{1 + e^{-lh}}{2}\right)^n =: \mathcal{W}(Y_0^h, Y_\infty^h) \times (V_{Euler})^n.\end{aligned}$$

Après une simple étude de fonction, on remarque que  $V_{Euler} \geq V_{TCP}$ , ce qui signifie que le TCP converge plus rapidement vers sa loi limite.

On regarde maintenant l'erreur de Wasserstein des lois limites. Soit  $t = nh$  un temps quelconque. On suppose que les deux processus partent du même point  $y > 0$  déterministe.

**Théorème 3.5.12.**

Pour tout  $h > 0$ , on a :

$$\mathcal{W}(Y_\infty, Y_\infty^h) \leq \frac{1}{1 - \alpha_h} \left\{ \beta_h + \frac{\gamma_h}{\ln\left(\frac{1 + e^{-lh}}{2}\right)} \left[ -1 + \ln\left(\frac{\gamma_h}{C_\infty^h (1 - \alpha_h) \ln\left(\frac{2}{1 + e^{-lh}}\right)}\right) \right] \right\}, \quad (3.5.5)$$

où  $C_\infty^h$  est une constante qui ne dépend que du point initial  $y$  et des moments d'ordre 1 et 2 des lois limites  $\pi$  et  $\pi^h$ .

**Preuve.** On a, par l'inégalité triangulaire :

$$\begin{aligned}\mathcal{W}(Y_\infty, Y_\infty^h) &\leq \mathcal{W}(Y_\infty, Y_{nh}) + \mathcal{W}(Y_{nh}, Y_{nh}^h) + \mathcal{W}(Y_{nh}^h, Y_\infty^h). \\ &\leq \mathcal{W}(Y_0, Y_\infty) e^{-\frac{1}{2}nh} + \mathbb{E}[|Y_{nh}^h - Y_{nh}|] + \mathcal{W}(Y_0^h, Y_\infty^h) \left(\frac{1 + e^{-lh}}{2}\right)^n \\ &\leq \left(\mathcal{W}(\delta_y, Y_\infty) + \mathcal{W}(\delta_y, Y_\infty^h)\right) \times \left(\frac{1 + e^{-lh}}{2}\right)^n + \mathbb{E}[|Y_{nh}^h - Y_{nh}|].\end{aligned}$$

Or, par l'inégalité de Jensen :

$$\begin{aligned}\mathcal{W}(\delta_y, Y_\infty) + \mathcal{W}(\delta_y, Y_\infty^h) &= \mathbb{E}[|y - Y_\infty|] + \mathbb{E}[|y - Y_\infty^h|] \leq \sqrt{y^2 + m_2 - 2ym_1} + \sqrt{y^2 + m_2^h - 2ym_1^h} \\ &:= C_\infty^h\end{aligned}$$

et par le théorème 3.4.1

$$\mathbb{E}[|Y_{nh}^h - Y_{nh}|] \leq \alpha_h^n \mathbb{E}[|Y_0^h - Y_0|] + \frac{1 - \alpha_h^n}{1 - \alpha_h} (\beta_h + n\gamma_h) \leq \frac{1}{1 - \alpha_h} (\beta_h + n\gamma_h).$$

Donc pour tout  $n \in \mathbb{N}^*$  :

$$\mathcal{W}(Y_\infty, Y_\infty^h) \leq C_\infty^h \left(\frac{1 + e^{-lh}}{2}\right)^n + \frac{\beta_h + n\gamma_h}{1 - \alpha_h} := \Psi_h(n). \quad (3.5.6)$$

Comme cette inégalité est valable pour tout  $n$ , on cherche à minimiser la fonction dérivable  $\Psi_h$ . Après quelques calculs, on trouve que sa dérivée est

$$\Psi'_h(n) = C_\infty^h \ln \left( \frac{1 + e^{-lh}}{2} \right) \exp \left( n \ln \left( \frac{1 + e^{-lh}}{2} \right) \right) + 0 + \frac{\gamma_h}{1 - \alpha_h},$$

donc

$$\Psi'_h(\tilde{n}) = 0 \iff \tilde{n} = \frac{1}{\ln \left( \frac{1 + e^{-lh}}{2} \right)} \ln \left( \frac{\gamma_h}{C_\infty^h (1 - \alpha_h) \ln \left( \frac{2}{1 + e^{-lh}} \right)} \right).$$

En réinjectant dans (3.5.6), on obtient

$$\begin{aligned} \Psi_h(\tilde{n}) &= C_\infty^h \left( \frac{\gamma_h}{C_\infty^h (1 - \alpha_h) \ln \left( \frac{2}{1 + e^{-lh}} \right)} \right) \\ &\quad + \frac{\beta_h}{1 - \alpha_h} + \frac{\gamma_h}{1 - \alpha_h} \times \frac{1}{\ln \left( \frac{1 + e^{-lh}}{2} \right)} \ln \left( \frac{\gamma_h}{C_\infty^h (1 - \alpha_h) \ln \left( \frac{2}{1 + e^{-lh}} \right)} \right) \\ &= \frac{\beta_h}{1 - \alpha_h} + \frac{\gamma_h}{1 - \alpha_h} \times \frac{1}{\ln \left( \frac{1 + e^{-lh}}{2} \right)} \left[ -1 + \ln \left( \frac{\gamma_h}{C_\infty^h (1 - \alpha_h) \ln \left( \frac{2}{1 + e^{-lh}} \right)} \right) \right]. \end{aligned}$$

□

**Remarque 3.5.13.** La constante  $C_\infty^h$  converge, quand  $h$  tend vers 0, vers  $2\sqrt{y^2 + m_2 - 2ym_1}$ . Elle est en particulier bornée. Lorsqu'on la majore par une constante qui ne dépend plus de  $h$ , un développement limité nous permet de voir que la formule (3.5.5) est en  $h \ln(1/h)$ .

### 3.5.3 Simuler cette erreur

Pour pouvoir simuler l'erreur de Wasserstein, on utilise la formule d'appariement aussi appelée *matching formula* :

#### Proposition 3.5.14.

Soit  $\mu$  et  $\nu$  deux mesures définies par

$$\mu = \frac{1}{n} \sum_{k=1}^n \delta_{x_k} \quad \text{et} \quad \nu = \frac{1}{n} \sum_{k=1}^n \delta_{y_k}$$

où  $x_1, \dots, x_n, y_1, \dots, y_n$  sont dans  $(H, |\cdot|)$ . Alors

$$\mathcal{W}(\mu, \nu) = \inf_{\sigma \in \mathfrak{S}_n} \frac{1}{n} \sum_{k=1}^n |x_k - y_{\sigma(k)}| \quad (3.5.7)$$

où  $\mathfrak{S}_n$  est l'ensemble des permutations d'ordre  $n$ .

La démonstration de cette formule est faite en page 5 de l'ouvrage [Vil09]. Cela nous permet d'avoir un moyen de simuler une approximation de la distance entre deux mesures de

probabilité  $\mu$  et  $\nu$  dans  $\mathcal{P}(\mathbb{R}, p)$ . Soit  $(X_i)_{i \geq 1}$  et  $(Y_i)_{i \geq 1}$  deux suites de variables aléatoires i.i.d. de loi  $\mu$  et  $\nu$  respectivement. Par la loi des grands nombres, on a convergence des mesures empiriques :

$$\frac{1}{n} \sum_{k=1}^n \delta_{X_k} \xrightarrow[n \rightarrow \infty]{\mathcal{W}_p} \mu \quad \text{et} \quad \frac{1}{n} \sum_{k=1}^n \delta_{Y_k} \xrightarrow[n \rightarrow \infty]{\mathcal{W}_p} \nu.$$

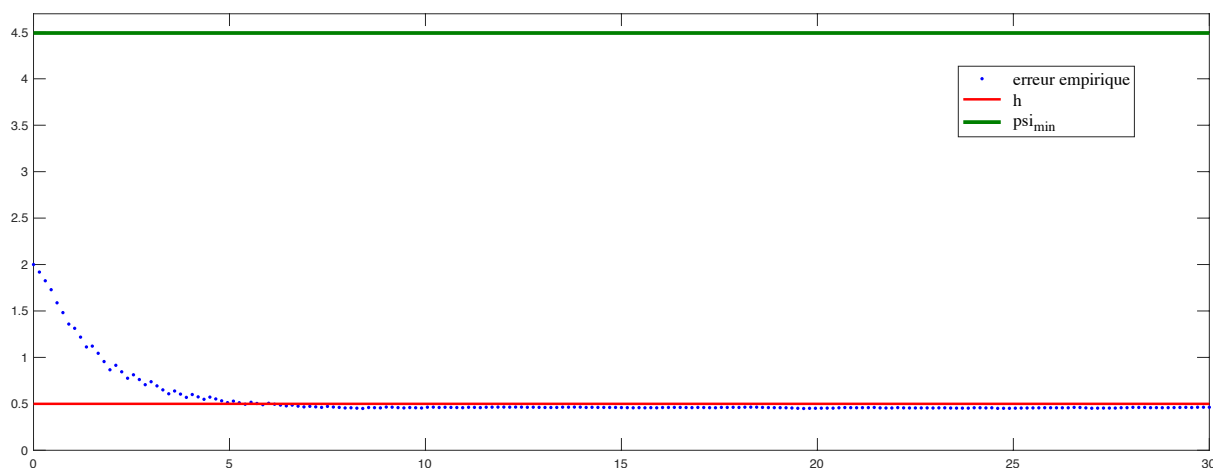
En combinant ces convergences avec la formule d'appariement (3.5.7), on obtient l'approximation suivante lorsque  $n$  tend vers l'infini :

$$\mathcal{W}_p(\mu, \nu)^p \simeq \inf_{\sigma \in \mathfrak{S}_n} \frac{1}{n} \sum_{k=1}^n |X_k - Y_{\sigma(k)}|^p.$$

Or si  $x, y, u, v$  sont des réels tels que  $x \leq y$  et  $u < v$  alors  $|x-v| + |y-u| \geq |x-u| + |y-v|$  : on minimise quand on compare "les plus petits" entre eux. Ainsi, on considère la permutation  $\sigma$  qui compare les échantillons ordonnés :

$$\mathcal{W}_p(\mu, \nu)^p \simeq \frac{1}{n} \sum_{k=1}^n |X_{(k)} - Y_{(k)}|^p.$$

Concernant le TCP, on fait la moyenne empirique de nos échantillons ordonnés.



**FIGURE 3.5.2** – *erreur de Wasserstein empirique et majorations théoriques.* On évalue l'erreur avec un pas  $\Delta s = 0.15$ , avec  $l = 1$ ,  $h = 0.5$ ,  $Y_0 = 5$ ,  $Y_0^h = 3$  jusqu'à  $t = 30$ .

Sur la figure 3.5.2, on remarque que l'erreur de Wasserstein empirique semble avoir un palier en  $h$ , alors que notre majoration théorique est assez au dessus. Pour le moment, les calculs n'ont pas permis d'obtenir une meilleure majoration.

## Conclusion

Tout au long de ce chapitre, on a présenté un PDMP particulier, très simple, sur lequel on peut déjà dégager des conclusion intéressantes.

D'abord, les majorations des différents types d'erreur et la simulation numérique qui en découle. Sur l'erreur forte, en combinant deux majorations on arrive à encadrer notre

erreur de façon non explosive et assez précise. Concernant l'erreur faible, la majoration est existante, bien que plus éloignée. On a de plus des développements limités des majorations de l'erreur lorsque le pas de notre schéma d'Euler tend vers 0, nous permettant de voir le comportement en fonction de  $h$ .

On donne ici quelques pistes de réflexion concernant les perspectives suite au travail développé dans ce chapitre.

- ❖ prendre l'intensité de saut qui ne serait pas constante. Par exemple, plus le temps avance, plus le débit a de chances d'être divisé par deux; cette généralisation impliquerait plusieurs changements :
- ✧ La simulation des durées ne serait plus aussi directe que l'utilisation du pseudo inverse de la fonction de répartition dans laquelle apparait l'intensité. Il existe des algorithmes de simulations via méthode du rejet (voir par exemple [Dev86], au chapitre II.3, ou encore l'article [Gil76]). Ces méthodes sont connues, mais dans le cadre des PDMP on ne sait pas l'effet sur les différentes erreurs d'approximations. On présente ci-dessous, sous forme de pseudo-code, l'algorithme 3.5.1 type Gillespie, présentant une façon de simuler un PDMP jusqu'à un certain temps donné  $T_{max}$ . Cet algorithme suppose qu'on connaît le flot et la fonction  $l \circ \phi$  et suppose aussi que cette fonction est bornée par une certaine quantité  $L$  connue. Il ne permet pas d'avoir une simulation exacte des temps de sauts lorsque le flot est non explicite, ce qui fait un décalage du processus pour ses positions post-saut.
- ✧ Les différentes durées inter-saut ne seraient plus identiquement distribuées, car leur fonction de répartition fait intervenir la fonction  $l \circ \phi(x, \cdot)$ , dépendant de la position  $x$  du processus au dernier saut. De plus, l'erreur commise par décalage des positions aux instants de saut se répercute dans le calcul des durées inter-saut à suivre, ce qui fait que l'erreur se cumule.

---

**Algorithm 3.5.1:** ALGOGILLESPIE( $T_{max}, L, x_0$ )

---

*on suppose que la fonction  $l \circ \phi$  est bornée par une constante  $L$*

$x \leftarrow x_0$

$t \leftarrow 0$

**tant que**  $t < T_{max}$

$$\left\{ \begin{array}{l} E \sim \mathcal{E}(L) \end{array} \right. \quad (1)$$

$$\left\{ \begin{array}{l} U \sim \mathcal{U}[0, L] \end{array} \right. \quad (2)$$

$$\left\{ \begin{array}{l} X_{t+s} \leftarrow \phi(x, s) \quad \forall s \in [0, E[ \end{array} \right. \quad (3)$$

$$\left\{ \begin{array}{l} t \leftarrow t + E \end{array} \right. \quad (4)$$

**faire**

$$\left\{ \begin{array}{l} \text{si} \quad U \leq l \circ \phi(x, E) \end{array} \right. \quad (5)$$

$$\left\{ \begin{array}{l} \text{alors} \quad x \sim \mathbf{Q}(\phi(x, E), \cdot) \end{array} \right. \quad (6)$$

$$\left\{ \begin{array}{l} \text{sinon} \quad x = \phi(x, E) \end{array} \right. \quad (7)$$


---

- ❖ prendre un flot non linéaire, dont l'approximation influence la valeur d'état mais également la probabilité de saut ; on aurait ici l'ajout d'une erreur issu du schéma. Les schémas numériques pour les équation-différentielles stochastiques, utilisés dans par exemple [KP92], exhibent déjà ces erreurs dans des cadres d'équations diffuses (faisant intervenir des mouvements brownien). Si on reprend l'algorithme 3.5.1, il faudrait remplacer tout les  $\phi$  par un  $\phi^h$ , qui serait un flot approchant le flot exact par schéma numérique (pour un pas  $h$  donné). On aurait alors une première erreur sur la trajectoire des morceaux déterministes, à la ligne (3), mais également dans le rejet ou non du saut (ligne (5)), et aussi pour la position post-saut (ligne (6)). Ce biais se répercuterait dans toutes les étapes de construction du processus, et la question de l'impact de cette erreur reste une question ouverte.
- ❖ se pose aussi la question de la grille fixe de points : en effet, pour faire l'approximation du TCP on s'est fixé une grille  $h\mathbb{N}$  à laquelle appartiennent les instants de saut du processus  $Y^h$ . Imposer les instants de saut à être sur une telle grille fixe, dans le cadre de modèle moins triviaux, ne semble pas raisonnable car on cumulerait une erreur supplémentaire. Ici, on a fait ce choix car le flot était linéaire et l'intensité constante, et donc l'erreur mesurée était uniquement due à ce décalage dans les instants de saut. En effet, utiliser un algorithme type Gillespie pour approcher ce modèle très simple de TCP ne nous apporterait aucune informations. Cependant pour des modèles plus compliqués, l'erreur supplémentaire due au flot approché et à la méthode de rejet nous fait envisager une grille "mobile", c'est-à-dire à chaque nouvel instants de saut, on décale la grille. On aurait donc une union du type

$$\bigcup_{n \geq 0} \left\{ T_n, T_n + h, T_n + 2h, \dots, T_n + h \left\lfloor \frac{T_{n+1}}{h} \right\rfloor \right\}$$

sur laquelle serait tous les points du processus  $Y^h$ .

# CONCLUSION ET PERSPECTIVES

En prenant comme objet d'étude les PDMP, nous avons étudié deux problèmes indépendants :

- ❖ Nous avons construits des PDMP sur des espaces de mesures, objets non encore introduits dans la littérature. Ces PDMP nous ont servi à modéliser des populations individu-centrées, c'est-à-dire dont la dynamique globale est construite par rapport à la dynamique d'un individu. La motivation de l'introduction de ces objets est, entre autre, un cadre plus agréable dans lequel manipuler ces objets : ainsi, à chaque saut la composante d'état de nos PDMP reste toujours dans l'espace  $\mathfrak{M}$  des mesures localement finies. De plus, étudier la structure de ces espaces de mesures nous a permis de voir que nous pouvions généraliser la construction de Davis, [Dav84]. De par leur propriété d'espace Polonais, il nous a en effet été possible de construire des variables aléatoires suivant la loi donnée par le noyau markovien du PDMP mesure.

Dans ce cadre de ces PDMP sur des espaces de mesure, nous avons défini un problème d'arrêt optimal avec un horizon aléatoire fini. L'extension de la programmation dynamique pour les PDMP à valeur mesure est inspirée de l'article de Gugerli [Gug86] pour les PDMP réels. Une suite logique de ce travail serait de s'intéresser à l'approximation numérique des fonctions valeurs, comme par exemple dans [dSDG10], où l'on utilise des techniques de quantification. Cependant, il reste le problème de la représentation numérique : en effet, ce sont des variables aléatoires à valeurs mesures et non des variables réelles. De plus, il serait intéressant de voir s'il est possible de faire du contrôle stochastique, et pas seulement de l'arrêt optimal ; c'est-à-dire que l'on pourrait intervenir plus d'une fois dans le changement de valeur forcée de notre processus.

- ❖ Nous nous sommes également intéressés à un problème d'approximation numérique sur un cas très simple, le TCP. Nous avons pu obtenir des majorations sur les écarts entre le processus TCP exact et le processus construit par schéma d'Euler entre les sauts. Nous avons également calculé des développements limités des dites majorations, afin d'avoir leur comportement en fonction du pas du schéma. Comme vu à la fin



du chapitre 3, quelques pistes ont déjà été abordées. On pourrait vouloir généraliser en prenant une intensité de saut non constante, ce qui changerait non seulement les techniques de simulation de tels processus, mais aussi cela rajouterait un décalage dans les instants de saut. On pourrait également prendre un flot non linéaire, et étudier l'erreur commise en faisant l'approche de ce flot par un schéma d'Euler.

Comme perspective concernant ces deux problèmes, on pourrait étudier un problème de contrôle stochastique sur un processus PDMP dont le flot ne serait pas connu explicitement mais via une équation différentielle. On retrouve ce genre de problématique dans par exemple [PPP11] dans le cadre de mathématiques financière, mais pour des processus diffusifs et non pas des PDMP.

## ANNEXE

### A

## CODES MATLAB

On présente dans cette annexe les algorithmes utilisés dans les différents chapitres de ce document, aussi les algorithmes de simulation de trajectoire, que les algorithmes pour le calcul des fonctions valeurs. Cette annexe est divisée en 3 parties, une pour chaque chapitre. De plus, il n'est pas présenté ici les commandes pour les titres et les légendes, afin de ne pas alourdir la présentation.

### A.1 Pour la construction générale

Il est présenté ici tous les algorithmes permettant différentes illustrations de résultats du chapitre de présentation des PDMP. Ils ont été créés sous MATLAB. Il n'est pas présenté ici les commandes pour les titres et les légendes, afin de ne pas alourdir la présentation.

**Listing A.1.0.1** – *simulation de l'évolution de la taille d'une cellule*

```
1 function une_cellule_simu(x,N,r,a,dt)
2     % x=vecteur des tailles initial
3     % N=nb de saut
4     % r=taux de croissance
5     % intensite l(x)=x^a
6     clf();
7     M=x;
8     T=0;
9     for i=1:N % boucle sur le nb de sauts
10        U=rand(1,1); %pour simuler tps d'inter-saut
11        S=(r*a)^(-1)*log(1-(r*a*log(U)/(M^a)));
12        % on utilise F^[-1](U) pour l'inter-tps
```

```

13     tau=T; %on enregistre la valeur du tps precedent
14     T=T+S;% nouveau temps de saut
15
16     y=[tau:dt:T,T]; %pour les graphiques
17
18         Y=M*exp(r*(y-tau)); % temps - precedent tps de saut
19         plot2d(y,Y);
20     M=M*exp(r*S); %phi(x,T_1)
21     plot2d([T,T],[0,M],style=5) % trait vertical
22     M=M/2;
23     end
24
25 end

```

## A.2 Pour l'optimisation

### A.2.1 Différence numérique entre les fonctions valeurs du modèle particule marquée et du modèle population

Listing A.2.1.1 – simulation de la fonction valeur pour l'individu centré

```

1  function v1=v1_x0(x,r,a,seuil,N,nbpt,nb)
2      % x : position initial
3      % r : taux de croissance
4      % a : intensite l(x)=x^a
5      % seuil : ou tronque-t-on notre identite
6      % N : taille echantillon dans M-C
7      % nbpt : nb de point sur l'intervalle de tps du calcul du sup
8      % nb : nb de realisation qu'on veut de v1
9
10     %on aura un vecteur avec plusieurs valeurs de v1
11
12     v1=zeros(1,nb);
13     for i=1:nb
14 %simulation l'echantillon entier de T1 (pour prendre le max)
15         U=rand(1,N); %pour simuler tps d'inter-saut
16         T1=(r*a)^(-1)*log(1-(r*a*log(U)/x^a));
17
18         Tmax=max(T1);
19
20         dt=Tmax/nbpt;
21         t=0:dt:Tmax;
22         nt=length(t);
23
24 %% calcul du premier terme dans v1(Z0) (le sup)
25
26         sup1=exp(-x^a*(exp(r*a*t)-1)/(a*r)).

```

```

27         *(seuil+(x*exp(r*t))-seuil).*(x*exp(r*t)<seuil));
28     %vecteur colonne de meme taille de t
29
30     %pour sup2, on doit calculer par monte-carlo
31     v0_J=zeros(N,nt);
32     v0_K=zeros(N,1);
33 %on doit faire "deux" fois le calcul de V0
34 %(pour I et pour K, les op de gugerli)
35     for k=1:N
36         v0_J(k,:)=(seuil+(x*exp(r*t))/2-seuil).
37                 *(x*exp(r*t)<2*seuil)).*(T1(k)<t);
38         v0_K(k)=(seuil+(x*exp(r*T1(k)))/2-seuil).
39                 *(x*exp(r*T1(k))<2*seuil));
40     end
41
42     sup2=mean(v0_J);
43 %vecteur ligne, cela nous donne la moyenne pour chaque valeur de t
44
45     sup=max(sup1+sup2);
46
47     v1(i)=max(sup,mean(v0_K));
48     end
49
50 end

```

**Listing A.2.1.2** – *simulation de la fonction valeur pour la vision mesure*

```

1
2     V1=zeros(1,nb);
3     n=length(x); %taille de la mesure initiale
4
5     p=x.^(a)/(sum(x.^a));
6     q=[0,cumsum(p)];
7
8     for i=1:nb
9
10        U=rand(1,N); %pour simuler tps d'inter-saut
11        T1=(r*a)^(-1)*log(1-(r*a*log(U)/(sum(x.^a))));
12
13        Tmax=max(T1);
14
15        dt=Tmax/nbpt;
16        t=0:dt:Tmax;
17        nt=length(t);
18
19 %calcul du premier terme dans V1(Z0) (le sup)
20
21        sup1=exp(-r*t-sum(x.^a)*(exp(r*a*t)-1)/(a*r)).
22        *(n*seuil+sum((x'*exp(r*t))-seuil).*(x'*exp(r*t)<seuil)));

```

```

23     %vecteur colonne de meme taille de t
24
25     %pour sup2, on doit calculer par monte-carlo
26     V0_J=zeros(N,nt);
27     V0_K=zeros(N,1);
28
29     for k=1:N %on fait notre M-C
30         u=rand(1,1);
31         i1=sum((q-u)<0); %simu num de la branche se divisant
32
33         som_i1=(x.*exp(r*T1(k))-seuil).*(x.*exp(r*T1(k))<seuil);
34         %pour faire une somme, mais sans i1
35         som_i1=[som_i1(1:i1-1),som_i1(i1+1:n)];
36
37         V0_J(k,:)=exp(-r*T1(k))*(sum(som_i1)+(x(i1)*
38         exp(r*T1(k))-2*seuil)*(x(i1)*exp(r*T1(k))<2*seuil)
39         +(n+1)*seuil)*(T1(k)<t);
40
41         V0_K(k)=exp(-r*T1(k))*(sum(som_i1)+(x(i1)*
42         exp(r*T1(k))-2*seuil)*(x(i1)*exp(r*T1(k))<2*seuil)
43         +(n+1)*seuil);
44     end
45
46     sup2=mean(V0_J);
47 %vecteur ligne, cela nous donne la moyenne pour chaque valeur de t
48
49     sup=max(sup1+sup2);
50
51
52     V1(i)=max(sup,mean(V0_K));
53 end
54 end

```

**Listing A.2.1.3** – plusieurs simulation des deux fonctions valeurs

```

1 nbb=length(500:500:10000);
2 M=zeros(3,nbb);
3 M(1,:)=1000:1000:100000;
4
5 x=3; N=10000; r=2; a=1; seuil=1; nbpt=1000;
6
7 M=zeros(2,50);
8 for i=1:50
9     M(1,i)=v1_x0(x,r,a,seuil,N,nbpt,1);
10    M(2,i)=V1_Z0(x,r,a,seuil,N,nbpt,1);
11 end
12
13 M([2,3],:)

```

## A.2.2 Illustration de $g(Z_1)$

Listing A.2.2.1 – simulation de valeurs de  $g(Z_1)$

```
1 function V0=V0_Z1(x,r,a,seuil,N)
2     % x=vecteur initial
3     % N=nb de simu qu'on veut de V0
4     % r=taux de croissance
5     % intensite l(x)=x^a
6     %on prend g(mu)=sum x_i exp(-rt) 1_{x_i<seuil}
7
8     % on a a la fin un N-echantillon de V0(Z1)
9
10    n=length(x); %taille de la mesure initiale
11    V_0=zeros(1,N);
12
13    p=x.^(a)/(sum(x.^a));
14    %la simulation de la branche choisie ne dep que du point de depart
15    %(pour lsaut), donc on garde la meme formule pour chaque simu
16    q=[0,cumsum(p)];
17
18    for k=1:N %on fait notre M-C
19        U=rand(1,1); %pour simuler tps d'inter-saut
20        T1=(r*a)^(-1)*log(1-(r*a*log(U)/(sum(x.^a))));
21        % on utilise F^[-1](U)
22        u=rand(1,1);
23        i1=sum((q-u)<0); % simu de la branche qui se divise
24
25        som_i1=(x.*exp(r*T1)-seuil).*(x.*exp(r*T1)<seuil);
26        %pour faire une somme, mais sans i1
27        som_i1=[som_i1(1:i1-1),som_i1(i1+1:n)];
28        V_0(k)=exp(-r*T1)*(sum(som_i1)+(x(i1)*exp(r*T1)-2*seuil)
29                *(x(i1)*exp(r*T1)<2*seuil)+(n+1)*seuil);
30    end
31
32    V0=V_0;
33 end
```

Listing A.2.2.2 – histogramme de  $g(Z_1)$

```
1 function V0=V0_Z1_hist(x,r,a,seuil,N)
2     % x=vecteur initial
3     % N=nb de saut -> ici N=1
4     % r=taux de croissance
5     % nb=nb de simu dans montecarlo
6     % intensite l(x)=x^a
7     %on prend g(mu)=sum x_i exp(-rt) 1_{x_i<seuil}
8
9     % la fonction renvoie l'histogramme de V0(Z0)
10    clf()
```

```

11     tic()
12     n=length(x); %taille de la mesure initiale
13     V_0=zeros(1,N);
14     %on veut plusieurs copies de V0(Z_1) pour avoir histogramme
15
16     p=x.^(a)/(sum(x.^a));
17     %la simulation de la branche choisie ne dep que du point de depart
18     %(pour lsaut), donc on garde la meme formule pour chaque simu
19     q=[0,cumsum(p)];
20
21     for k=1:N %on fait notre M-C
22         U=rand(1,1); %pour simuler tps d'inter-saut
23         T1=(r*a)^(-1)*log(1-(r*a*log(U)/(sum(x.^a))));
24
25         u=rand(1,1);
26         i1=sum((q-u)<0); % simu de la branche qui se divise
27
28         som_i1=(x.*exp(r*T1)-seuil).*(x.*exp(r*T1)<seuil);
29         %pour faire une somme, mais sans i1
30         som_i1=[som_i1(1:i1-1),som_i1(i1+1:n)];
31         V_0(k)=exp(-r*T1)*(sum(som_i1)+(x(i1)*exp(r*T1)-2*seuil)
32             *(x(i1)*exp(r*T1)<2*seuil)+(n+1)*seuil);
33     end
34     % on a des simu de Z_1
35     toc()
36     V0=mean(V_0);
37     h=histogram(V_0,'Normalization','pdf');
38     hold on;
39 end

```

**Listing A.2.2.3** – *histogramme de  $g(Z_1)$  pour deux cellules initiales avec densités théoriques*

```

1 function V0=V0_2_Z1_hist(x,r,a,seuil,N)
2
3     x=sort(x,'descend');
4
5     % la fonction renvoie l'histogramme de V0(Z1)
6     % avec densite theorique correspondante
7     clf()
8     tic()
9     n=length(x);
10    %taille de la mesure initiale, seulement deux cellules
11
12    V_0=zeros(1,N); %plein de valeur pour faire hist
13
14    p=x.^(a)/(sum(x.^a));
15    q=[0,cumsum(p)];
16    X_a=sum(x.^a);
17    coeff=seuil^a*exp(X_a/(r*a))/r;

```

```

18     x_2=x(2);
19     x_1=x(1);
20
21     for k=1:N %on fait notre M-C
22         U=rand(1,1); %pour simuler tps d'inter-saut
23         T1=(r*a)^(-1)*log(1-(r*a*log(U)/(sum(x.^a))));
24         % on utilise F^[-1](U)
25
26         u=rand(1,1);
27         % pour simuler le num de la branche qui se divise
28         i1=sum((q-u)<0);
29         x_n=[x(1:i1-1),x(i1+1:2)];
30
31         V_0(k)=3*seuil*exp(-r*T1)+(x_n-seuil*exp(-r*T1))*
32             (x_n<seuil*exp(-r*T1))+(x(i1)-2*seuil*exp(-r*T1))*
33             (x(i1)<2*seuil*exp(-r*T1));
34     end
35     % on a des simu de Z_1
36     toc()
37     V0=mean(V_0);
38     h=histogram(V_0,'Normalization','pdf');
39
40
41     if seuil<=(min(x)/2) %turquoise
42         ecart=max(V_0)-min(V_0);
43         xx=min(V_0):ecart/1000:max(V_0);
44     %on trace la densite entre les valeurs min/max de l'echantillon
45     xx_d=((n+1)^a*seuil^a*X_a)/(r*a*xx.^a);
46     densite=exp(X_a/(r*a))*a./xx.*xx_d.*exp(-1.*xx_d).
47     *(xx<3*seuil);
48     hold on;
49     plot(xx,densite,'color',[0,1,1]);
50     plot([(n+1)*seuil,(n+1)*seuil],[0,densite(end)]);
51     %on trace lignes horizontales au niveau
52     %des bornes des indicatrices theoriques
53     plot([0,0],[0,max(densite)],'color',[0,1,1]);
54     plot([3*seuil,3*seuil],[0,max(densite)],'color',[0,1,1]);
55     end
56
57     if ((x_2/2)<seuil)&&(seuil<=min(x_2,x_1/2)) %rose
58         ecart=max(V_0)-min(V_0);
59         xx=min(V_0):ecart/1000:max(V_0);
60         % xx_d=(3^a*seuil^a*X_a)/(r*a*xx.^a);
61         densite=coeff*((n+1)^a./xx.^(a+1)).
62         *exp(-(3^a*seuil^a*X_a)/(r*a.*xx.^a)).*(x_1^a.
63         *(xx<3*seuil)+ x_2^a.*(xx<1.5*x_2))+
64         x(2)^a./(xx-x(2)).^(a+1).
65         *exp(-(seuil^a*X_a)/(r*a.*(xx-x(2)).^a)).
66         *(1.5*x_2<xx).*(xx<(x_2+seuil)));

```



```

67     hold on;
68     plot(xx,densite,'color',[1,0,1]);
69     plot([(n+1)*seuil,(n+1)*seuil],[0,densite(end)]);
70     plot([1.5*x_2,1.5*x_2],[0,max(densite)]);
71     plot([seuil+x_2,seuil+x_2],[0,max(densite)],);
72     plot([0,0],[0,max(densite)],'color',[1,0,1]);
73     plot([3*seuil,3*seuil],[0,max(densite)]);
74 end
75
76 if (x_2<seuil)&&(seuil<=(x_1/2)) %marron
77     ecart=max(V_0)-min(V_0);
78     xx=min(V_0):ecart/1000:max(V_0);
79     %xx_d=(3^a*seuil^a*X_a)/(r*a*xx.^(a));
80     densite=coeff.*(2^a*x_1^a./(xx-x_2).^(a+1)).*
81     exp(-2^a*seuil^a*X_a/(r*a*(xx-x_2).^a)).
82     *(3*x_2<xx).*(xx<x_2+2*seuil)+3^a*
83     x_1^a./(xx).^(a+1).*exp(-3^a*seuil^a*X_a/(r*a*(xx).^a)).
84     *(0<xx).*(xx<3*x_2)+x_2^a./(xx-x_2).^(a+1).
85     *exp(-seuil^a*X_a/(r*a*(xx-x_2).^a)).*(1.5*x_2<xx).
86     *(xx<x_2+seuil)+3^a*x_2^a./(xx).^(a+1).
87     *exp(-3^a*seuil^a*X_a/(r*a*(xx).^a)).
88     *(0<xx).*(xx<1.5*x_2));
89     hold on;
90     plot(xx,densite,'color',[0.5,0.25,0.1]);
91     plot([x_2+2*seuil,x_2+2*seuil],[0,densite(end)]);
92     plot([1.5*x_2,1.5*x_2],[0,max(densite)],);
93     plot([seuil+x_2,seuil+x_2],[0,max(densite)]);
94     plot([3*x_2,3*x_2],[0,max(densite)],);
95     plot([x_2+2*seuil,x_2+2*seuil],[0,max(densite)],);
96 end
97
98 if (x_2<=(x_1/2))&&(seuil>(x_1/2))&&(seuil<x_1) %rouge
99     ecart=max(V_0)-min(V_0);
100    xx=min(V_0):ecart/1000:max(V_0);
101    %xx_d=(3^a*seuil^a*X_a)/(r*a*xx.^(a));
102    densite=coeff.*(1./(xx-x_2).^(a+1)).*(2^a*x_1^a.
103    *exp(-(2^a*seuil^a*X_a)/(r*a*(xx-x_2).^a)).
104    *(3*x_2<xx).*(xx<(x_2+x_1))+x(2)^a.
105    *exp(-(seuil^a*X_a)/(r*a*(xx-x(2)).^a)).
106    *(1.5*x_2<xx).*(xx<(x_2+seuil)))
107    +3^a*seuil^a/(r*xx.^(a+1)).
108    *exp(X_a/(r*a)).*exp(-3^a*seuil^a*X_a.
109    /(r*a*xx.^a)).*(x_1^a.*(0<xx).
110    *(xx<3*x_2)+x_2^a.*(0<xx).*(xx<1.5*x_2));
111    densite(end)=densite(end)+x_1^a*(1-exp(X_a/(r*a))
112    *exp(-2^a*seuil^a*X_a/(r*a*x_1^a)))/X_a;
113    hold on;
114    plot(xx,densite,'color',[1,0,0]);
115    plot([x_2+x_1,x_2+x_1],[0,densite(end)],'color',[1,0,0],);

```

```

116     plot([1.5*x_2,1.5*x_2],[0,max(densite)]);
117     plot([seuil+x_2,seuil+x_2],[0,max(densite)]);
118     plot([3*x_2,3*x_2],[0,max(densite)]);
119     plot([x_2+x_1,x_2+x_1],[0,max(densite)]);
120     end
121
122     if (x_2<(x_1/2))&&(seuil>=x_1) %vert
123         ecart=max(V_0)-min(V_0);
124         xx=min(V_0):ecart/1000:max(V_0);
125         %xx_d=(3^a*seuil^a*X_a)/(r*a*xx.^(a));
126         densite=coeff.*(1./(xx-x_2).^(a+1)).
127         *(2^a*x_1^a.*exp(-(2^a*seuil^a*X_a).
128         /(r*a.*(xx-x_2).^a)).*(3*x_2<xx).
129         *(xx<(x_2+x_1))+x(2)^a.*exp(-(seuil^a*X_a).
130         /(r*a.*(xx-x(2)).^a)).*(1.5*x_2<xx).*(xx<(x_2+x_1)));
131         densite(end)=densite(end)+x_1^a*(1-exp(X_a/(r*a))
132         *exp(-2^a*seuil^a*X_a/(r*a*x_1^a)))/X_a+x_2^a
133         *(1-exp(X_a/(r*a))*exp(-seuil^a*X_a/(r*a*x_1^a)))/X_a;
134         hold on;
135         plot(xx,densite,'color',[0,1,0],'linewidth',2);
136         plot([x_2+x_1,x_2+x_1],[0,densite(end)'],'color',[0,1,0]);
137         plot([1.5*x_2,1.5*x_2],[0,max(densite)'],'color',[0,1,0]);
138         plot([3*x_2,3*x_2],[0,max(densite)'],'color',[0,1,0]);
139         plot([x_2+x_1,x_2+x_1],[0,max(densite)'],'color',[0,1,0]);
140     end
141
142     if ((x_2/2)<=(x_1/2))&&((x_1/2)<seuil)&&(seuil<x_2) %bleu
143         ecart=max(V_0)-min(V_0);
144         xx=min(V_0):ecart/1000:max(V_0);
145         %xx_d=(3^a*seuil^a*X_a)/(r*a*xx.^(a));
146         densite=coeff.*(x_1^a./((xx-x_1).^(a+1)).
147         *exp(-(seuil^a*X_a)/(r*a.*(xx-x_1).^a)).
148         *(1.5*x_1<xx).*(xx<(x_1+seuil))+x_2^a.
149         /((xx-x_2).^(a+1)).*exp(-(seuil^a*X_a).
150         /(r*a.*(xx-x_2).^a)).*(1.5*x_2<xx).*(xx<(x_2+seuil)));
151         hold on;
152         plot(xx,densite,'color',[0,0,1],'linewidth',2);
153         plot([seuil+x_1,seuil+x_1],[0,densite(end)]);
154         plot([1.5*x_1,1.5*x_1],[0,max(densite)]);
155         plot([seuil+x_2,seuil+x_2],[0,max(densite)]);
156         plot([1.5*x_2,1.5*x_2],[0,max(densite)]);
157         plot([seuil+x_1,seuil+x_1],[0,max(densite)]);
158     end
159
160     if ((x_1/2)<x_2)&&(x_2<=seuil)&&(seuil<x_1) %orange
161         ecart=max(V_0)-min(V_0);
162         xx=min(V_0):ecart/1000:max(V_0);
163         %xx_d=(3^a*seuil^a*X_a)/(r*a*xx.^(a));
164         densite=coeff.*(x_1^a./((xx-x_1).^(a+1)).

```

```

165 *exp(-(seuil^a*X_a)/(r*a.*(xx-x_1).^a)).*(1.5*x_1<xx).
166 *(xx<(x_1+x_2))+x_2^a./((xx-x_2).^(a+1)).
167 *exp(-(seuil^a*X_a)/(r*a.*(xx-x_2).^a)).*(1.5*x_2<xx).
168 *(xx<(x_2+seuil));
169 densite(end)=densite(end)+x_1^a*(1-exp(X_a/(r*a))
170 *exp(-seuil^a*X_a/(r*a*x_2^a)))/X_a;
171 hold on;
172 plot(xx,densite,'color',[1,0.75,0.5],'linewidth',2);
173 plot([x_2+x_1,x_2+x_1],[0,densite(end)]);
174 plot([1.5*x_1,1.5*x_1],[0,max(densite)]);
175 plot([seuil+x_2,seuil+x_2],[0,max(densite)]);
176 plot([1.5*x_2,1.5*x_2],[0,max(densite)]);
177 plot([x_2+x_1,x_2+x_1],[0,max(densite)]);
178 end
179
180 if ((x_1/2)<x_2)&&(x_2<x_1)&&(x_1<=seuil) %violet
181 ecart=max(V_0)-min(V_0);
182 xx=min(V_0):ecart/1000:max(V_0);
183 %xx_d=(3^a*seuil^a*X_a)/(r*a*xx.^a);
184 densite=coeff.*(x_1^a./((xx-x_1).^(a+1)).
185 *exp(-(seuil^a*X_a)/(r*a.*(xx-x_1).^a)).*(1.5*x_1<xx).
186 *(xx<(x_1+x_2))+x_2^a./((xx-x_2).^(a+1)).
187 *exp(-(seuil^a*X_a)/(r*a.*(xx-x_2).^a)).*(1.5*x_2<xx).
188 *(xx<(x_2+x_1)));
189 densite(end)=densite(end)+x_1^a*(1-exp(X_a/(r*a))
190 *exp(-seuil^a*X_a/(r*a*x_2^a)))/X_a+x_2^a
191 *(1-exp(X_a/(r*a))*exp(-seuil^a*X_a/(r*a*x_1^a)))/X_a;
192 hold on;
193 plot(xx,densite,'color',[0.75,0.5,1],'linewidth',2);
194 plot([x_2+x_1,x_2+x_1],[0,densite(end)]);
195 plot([1.5*x_2,1.5*x_2],[0,max(densite)]);
196 plot([1.5*x_1,1.5*x_1],[0,max(densite)]);
197 plot([x_2+x_1,x_2+x_1],[0,max(densite)]);
198 end
199 end

```

## A.3 Pour le TCP

Il est présenté ici tous les algorithmes permettant différentes illustrations de résultats de la partie TCP. Ils ont été créés sous MATLAB.

### A.3.1 Construction de processus TCP

Listing A.3.1.1 – valeurs d'un TCP aux instants de sauts

```

1 function [Yg,Yd,E] = TCPsimu(y,t,l)
2 % y=position initiale,

```

```

3 %t=temps que l'on veut regarder ,
4 %l=parametre de l'exp
5 % sortie = vecteur des etats (g et d) + instant des sauts
6 E=[];
7 tau=0;
8 while tau<=t % tant qu'on n'a pas depasse t, on continue de sauter
9     U=rand(1,1); % on genere un nouveau inter-temps de saut
10    Ex=-log(1-U)/l; %pour avoir une loi expo
11    E=[E,Ex]; % vecteur des inter-temps
12    tau=tau+Ex; % c'est le dernier instant de saut
13 end
14
15 n=length(E); % (nb de sauts avant t) +1
16 %(car a la fin de la boucle, on a forcement depasse t)
17 Yg=[y]; % on fait maintenant les positions :
18 %limites gauche et les limites a droites
19 % => (donc deux valeurs pour chaque sauts)
20 yg=y;
21 Yd=[y];
22 yd=y;
23 i=1;
24 while i<=n
25     yg=yd+E(i); % limite a gauche
26     Yg=[Yg,yg];% vecteur total des limite a gauche
27     yd=yg/2; % limite a droite : on divise par deux
28     Yd=[Yd,yd]; % vecteur total des limite a droites
29     i=i+1; % on monte d'un cran
30 end
31
32 end

```

**Listing A.3.1.2** – évaluer la valeur d'un TCP à un instant  $s$

```

1 function ys=TCPvalue(Yg,Yd,E,s)
2 % entree = procesus caracterise par les instants de sauts et
3 % leurs limites gauche/droite
4 T=[0,cumsum(E)]; %instant des sauts avec etat initial
5     if (s>T(end)) then ys='erreur'
6         disp('le processus ne va pas jusqu a ce point')
7     end %il ne faut pas que s depasse le dernier temps
8 i=1;
9     while (s>=T(i))
10         i=i+1;
11     end % s \in [T(i-1) ; T(i)[ car T(1)=T0=0
12     %on cherche dans quel "inter-temps" la valeur s est.
13
14 ys=Yd(i-1)+s-T(i-1); %valeur du procesus en s
15 end

```

Listing A.3.1.3 – Graphique du TCP (juste à partir de paramètres)

```

1 function []=TCPg(y,t,l)
2 %sortie = graphe du processus jusqu'au saut apres t
3     clf()
4     tau=0;
5     E=[];
6
7     while tau<=t
8         U=rand(1,1);
9         Ex=-log(1-U)/l;
10        E=[E,Ex];
11        tau=tau+Ex;
12    end
13
14    T=[0,cumsum(E)];
15    % vecteur des instants de sauts (pour le graphique)
16
17    n=length(E);
18    Yg=[y];
19    yg=y;
20    Yd=[y];
21    yd=y;
22    i=1;
23    while i<=n
24        yg=yd+E(i);
25        Yg=[Yg,yg];
26        yd=yg/2;
27        Yd=[Yd,yd];
28        i=i+1;
29    end
30    %on trace le graphique
31    for i=1:n
32        hold on;
33        plot([T(i),T(i+1)],[Yd(i),Yg(i+1)])
34        hold on;
35        plot([T(i+1),T(i+1)],[0,Yg(i+1)],'m--') % trait verticaux
36        %pour mettre les traits en pointille aux instants de sauts
37    end
38    plot(T',Yg','o b')
39    plot(T',Yd','. b')
40    %l'instant de saut avant t est en n-1, ie en n pour T et Yd
41    % car on a rajoute 0 (qui correspond a l'etat y)
42    plot(cumsum(E),0*E, '+ m') %on affiche les instants de sauts
43 end

```

## A.3.2 Approche par schéma d'Euler

**Listing A.3.2.1** – Graphique du TCP et schéma d'Euler (juste à partir de paramètres) par méthode "partie entière"

```

1 function m=floorTCPg(y,yh,t,l,h)
2 clf()
3 [Yg,Yd,E] = TCPsimu(y,t,l);
4 T=[0,cumsum(E)];
5
6 for i=1:n
7     hold on;%pour tracer plusieurs courbes avec la mm echelle
8     plot([T(i),T(i+1)],[Yd(i),Yg(i+1)], 'linewidth',2);
9     %on trace les pentes
10 end
11 % l'instant de saut avant t est en n-1, ie en n pour T et Yd
12 %car on a rajoute 0 (qui correspond a l'etat y)
13 m=Yd(n)+t-T(n);
14 % on enleve l'avant-dernier saut, ie le saut avant t
15 hold on;
16 plot(t,m,'. g'); % la valeur du processus en t
17 hold on;
18 plot(T,0*T,'+ b'); % les instants de saut sur l'axe temporel
19
20 %on fait l'approximation :
21 Nh=floor(E/h)+1; % on calcule le nb de cran entre chaque saut
22 Mh=cumsum(Nh);
23 Yh=[yh+h*(0:Nh(1)-1)]; % on fait le premier escalier a part
24 % car on commence par y et pas par un saut (qui ferait y/2)
25 hold on;
26 plot(h*Mh,0*Mh,'+ r')
27 hold on;
28 stairs(h*(0:Nh(1)-1),Yh,'red','linewidth',1.5)
29 i=2;
30 while i<=n
31     Yh=[Yh(end)/2+h*(0:Nh(i)-1)];
32     hold on;
33     stairs(h*(Mh(i-1):Mh(i)-1),Yh,'red','linewidth',1.5);
34     i=i+1;
35 end
36
37 end

```

**Listing A.3.2.2** – Graphique du TCP et schéma d'Euler (juste à partir de paramètres) par méthode "fonction inverse"

```

1 function [m,EE]=unifTCPg(y,yh,t,l,h)
2 %sortie = valeur en t et vecteur des intersauts
3     clf
4     N=30*floor(1/h);
5     tau=0;
6     E=[];

```

```

7     Nh=[];
8     p=1-exp(-1*h); % c'est le parametre des geometriques
9     % simu d'une G(p) via U[0,1]
10    z=p*(1-p).^([0:N-1]);
11    z=[z,1-sum(z)];%comme support infini,
12    %on fait juste 29 classes puis la classe qui rassemble le reste
13    F=[0,cumsum(z)];%fonction de repartition
14
15    while tau<=t % tant qu'on n'a pas depasse t, on saute
16        U=rand(1,1); %loi unif sur [0,1]
17        Ex=-log(1-U)/1; % loi expo a partir de U
18        E=[E,Ex];
19        tau=tau+Ex; % vecteur des inter-temps
20        Nhh=dsearch(U,F); % on a la geometrique
21        Nh=[Nh,Nhh]; % vecteur des crans entre chaque saut
22    end
23
24    T=[0,cumsum(E)];
25    n=length(E);
26    Yg=[y];
27    yg=y;
28    Yd=[y];
29    yd=y;
30    i=1;
31    while i<=n
32        yg=yd+E(i);
33        Yg=[Yg,yg];
34        yd=yg/2;
35        Yd=[Yd,yd];
36        i=i+1;
37    end
38
39    for i=1:n
40        hold on;
41        plot([T(i),T(i+1)], [Yd(i),Yg(i+1)]);
42    end
43
44    m=Yd(n)+t-T(n);
45    EE=E;
46    hold on;
47    plot(t,m,'.g');
48
49    Mh=cumsum(Nh);
50    Yh=[yh+h*(0:Nh(1))]; plot2d2(h*(0:Nh(1)),Yh,style=5)
51    i=2;
52    while i<=n
53        Yh=[Yh(end)/2+h*(0:Nh(i))];
54        plot2d2(h*(Mh(i-1):Mh(i)),Yh,style=5);
55        i=i+1;

```

```

56     end
57
58 end

```

**Listing A.3.2.3** – Schéma d'Euler à partir d'un processus TCP

```

1
2 function [Yh,Mh] = EulerTCP(Yg,Yd,E,yh,h)
3 % fonction qui construit le schema d'Euler a partir d'un TCP
4 % caracterise par [Yg,Yd,E]
5 %h=pas du schema,
6 % yh=valeur initiale
7 T=cumsum(E); %instants des sauts
8
9     %on fait l'approximation :
10    Nh=floor(E/h)+1; %ce sont les 'inter-crans' des instants
11                    % de sauts sur la discretisation de pas h
12    Mh=cumsum(Nh);
13    Sh=h*Mh; %on a h x des geometrique (=crans de sauts)
14    n=length(E);
15    Yh=[yh+h*(0:Nh(1)-1)];
16    i=2;
17    while i<=n
18        Xh=[Xh,Xh(end)/2+h*(0:Nh(i)-1)];
19        i=i+1;
20    end
21 end

```

**Listing A.3.2.4** – Tracer le graphique du TCP avec son schéma d'Euler (ou non)

```

1 function graphe(Yg,Yd,E,yh,h)
2     clf();
3     l=1/mean(E);
4 %on a une approximation de lambda (tres mauvaise si peu de sauts)
5     y=Yg(1);
6     f=gcf();
7     T=[0,cumsum(E)]; % vecteur des instants de sauts
8     n=length(E);
9     for i=1:n
10        hold on;%pour tracer plusieurs courbes avec la mm echelle
11        plot([T(i),T(i+1)],[Yd(i),Yg(i+1)])
12    end
13    hold on
14    plot(T',Yd','b')
15    if (h==0); % si on choisit h=0 alors cela ne trace que le TCP
16        for i=1:n
17            hold on;
18            plot([T(i+1),T(i+1)],[0,Yg(i+1)],'m--')
19            hold on

```



```

20         plot(T',Yg','o b')
21     end
22     else
23         Nh=floor(E/h)+1;
24         Mh=cumsum(Nh);
25         Yh=[yh+h*(0:Nh(1)-1)];
26         hold on;
27         plot(h*Mh,0*Mh,'+ r')
28         hold on;
29         stairs(h*(0:Nh(1)-1),Yh,'red')
30         i=2;
31         while i<=n
32             Yh=[Yh(end)/2+h*(0:Nh(i)-1)];
33             hold on;
34             stairs(h*(Mh(i-1):Mh(i)-1),Yh,'red');
35             i=i+1;
36         end
37     end
38 end

```

Listing A.3.2.5 – *histogramme des lois "limites"*

```

1 function []=hist_lois_limites(l,h,n,x,xh,N,nbins)
2 %l=lambd, h=pas du schema, n=cran qu'on regarde (qu'on veut faire
3 %tendre vers l'infini
4 %N=nb de simu
5 %nbins=nb de classe dans l'histogramme
6 clf
7 xxh=abs(x-xh);
8 Y=1:N; %on va rentrer N simulations de la valeur en t_max=n*h
9 Yh=1:N;
10 t=n*h;
11
12 for i=1:N
13     tau=0;
14     tauh=0;
15     xg=x;
16     xd=x;
17     xhg=xh;
18     xhd=xh;
19 %besoin de la derniere valeur => ne garde pas toutes les valeurs
20
21     while tau<t
22         U=rand(1,1);
23         E=-log(1-U)/l;
24         tau=tau+E;
25         xg=xd+E;
26         xd=xg/2;
27

```

```

28         if tauh<t %on s'assure que EULER s'arrete aussi en t,
29         %sinon il irait plus loin car valeur entiere superieure
30             Nh=floor(E/h)+1;
31             Eh=h*Nh;
32             tauh=tauh+Eh;
33             xhg=xhd+h*(Nh-1);
34             xhd=xhg/2;
35         end
36
37     end
38     %a la fin, on a acces a E (inter-temps qui "contient" t),
39     %xg et xd (qui sont apres t), insant tau de saut apres t,
40     % et de meme pour euler (Eh, tauh, xgh, xdh)
41
42     Y(i)= xg-(tau-t); %comme on est apres t, il faut "reculer"
43     Yh(i)= xhg+h*h*floor(t/h)-tauh;
44 end
45 YY=zeros(N,2);
46 YY(:,1)=Y;
47 YY(:,2)=Yh;
48 C=hist(YY,nbins);
49 C(:,1)=C(:,1)./sum(C(:,1));%pour avoir normalisation
50 C(:,2)=C(:,2)./sum(C(:,2));
51 bar(C);
52 end

```

### A.3.3 Estimation erreur forte

Listing A.3.3.1 – estimation de l'erreur forte

```

1 function [Sup,ESP]=espTCP(y,yh,t,l,h,N,s)
2 % y,yh=positions initiales,
3 %t=temps total que l'on veut regarder,
4 % l=parametre de l'exp
5 % h=pas du schema;
6 % N nb de simu qu'on fait pour avoir une moyenne empirique;
7 % s=pas de temps qu'on regarde les differences entre les 2 proc
8 %en sortie : le sup des moyennes sur [0,t] ; le vecteur valeur
9     clf
10    S=0:s:t;
11    ecartinital=abs(y-yh);
12    Esp=zeros(N,length(S));
13 % on fait moyenne empirique en plusieurs valeurs de s,
14 %=>on simule plusieurs fois nos courbes (dans les lignes)
15 % puis on les evalue en plusieurs s (colonnes)
16 %=>chaque colonne correspond a N simulations evaluees en un s fixe
17    for j=1:N %lignes qui varient pour remplir les colonnes)
18        [Yg,Yd,E]=TCPsimu(y,t,l);

```

```

19     T=cumsum(E);
20     Nh=floor(E/h)+1;
21     Mh=cumsum(Nh);
22     Sh=h*Mh;
23     n=length(E);
24     Yh=[yh+h*(0:Nh(1)-1)];
25     i=2;
26     while i<=n
27         Yh=[Yh,Yh(end)/2+h*(0:Nh(i)-1)];
28         i=i+1;
29     end
30     for k=1:length(S);
31         %colonnes qui varient (pour remplir les lignes)
32         Esp(j,k)=abs(TCPvalue(Yg,Yd,E,S(k))
33                     -Yh(floor(S(k)/h)+1));
34     end
35 end
36 ESP=mean(Esp);
37 %ESP(j)=mean(Esp(:,j)) =moyenne au temps s
38 %(c'est un vecteur ligne qui fait la moyenne des colonnes de Esp)
39 Sup=max(ESP);
40 plot(S,ESP,'+ r','linewidth',2)
41 % moyenne de l'erreur en fonction du temps
42 end}

```

### A.3.4 Estimation erreur faible

Listing A.3.4.1 – estimation de l'erreur faible (d'ordre  $p$ ) en fonction du temps

```

1 function [w] = wassersteinp(y,yh,t,l,h,N,s,p)
2 %on veut simuler la distance de Wasserstein d'ordre p des lois de
3 %Y_t et Y_t^h, pour des instants de pas s entre 0 et t.
4
5 clf
6 yyh=abs(y-yh);
7 S=0:s:t;
8 Yn=zeros(N,length(S));
9 %les colonnes = N-echantillon de la valeur en j*s
10 Yhn=zeros(N,length(S));
11 for i=1:N %lignes qui varient pour remplir les colonnes :
12 %la colonne C_k contient un N-echantillons du TCP a la valeur k*s
13     [Yg,Yd,E]=TCPsimu(y,t,l);
14     [Yh,Mh] = EulerTCP(Yg,Yd,E,yh,h)
15     % on fait les vecteurs discretises pour les ordonner et
16     % prendre la moyenne empirique de l'erreur
17     for j=1:length(S);
18         Yn(i,j)=TCPvalue(Xg,Xd,E,S(j));
19         Yhn(i,j)=Xh(floor(S(j)/h)+1);

```

```

20     end
21 end
22 A=sort(Yn); % ca trie en ordre croissant les colonnes
23 %-> on ordonne les N-echantillons pour chaque valeurs j*s
24 Ah=sort(Yhn);
25
26 B=abs(A-Ah).^p; %on fait la |...|^p de la difference des
27 %echantillons ordonnes : B(i,j)=|Y^(i)_{j*s}-Y^{h,(i)}_{j*s}|^p
28 y=(mean(B)).^(1/p);
29 %moyenne empirique selon les colonnes ie y(j)=mean(B(:,j))
30 %=(1/N sum_{1?i?N}|Y^(i)_{j*s}-Y^{h,(i)}_{j*s}|^p)^1/p
31 plot(S,y, '+ b');
32 end

```

**Listing A.3.4.2** – estimation de l'erreur faible avec majoration théorique en fonction du temps

```

1 function [w] = wasserstein(x,t,l,h,N,s)
2 %on veut simuler la distance de Wasserstein des lois de Y_t et
3 % Y_t^h, pour des instants de pas s entre 0 et t, avec N simu.
4 clf
5 S=0:s:t;
6 Yn=zeros(N,length(S)); %comme d'ans l'algorithme precedent
7 Yhn=zeros(N,length(S));
8 for i=1:N
9     [Xg,Xd,E]=TCPsimu(x,t,l);
10    T=cumsum(E);
11
12    Nh=floor(E/h)+1;
13    Mh=cumsum(Nh);
14    Sh=h*Mh;
15    n=length(E);
16    Xh=[x+h*(0:Nh(1)-1)];
17    k=2;
18    while k<=n
19        Xh=[Xh,Xh(end)/2+h*(0:Nh(k)-1)];
20        k=k+1;
21    end
22
23    for j=1:length(S);
24        Yn(i,j)=TCPvalue(Xg,Xd,E,S(j));
25        Yhn(i,j)=Xh(floor(S(j)/h)+1);
26    end
27 end
28 A=sort(Yn);
29 Ah=sort(Yhn);
30
31 B=abs(A-Ah);
32 y=mean(B);

```

```

33 plot(S,y,'+ b','linewidth',2);
34 hold on
35 plot(S,h*ones(1,length(S)),'red','linewidth',1.5)
36
37 %on veut la mahoration theorique
38 alphah=(1+3*exp(-1*h)+1*h*exp(-1*h))/4;
39 rhoh=1/4*1^2*h^3;
40 betah=1/4*1^2*h^2*x
41 +1*h^2*(1+(3/8)*exp(-1*h))-5*(1-exp(-1*h)-1*h*exp(-1*h))/(4*1);
42 gammah=log((1+exp(-1*h))/2);
43 cte=sqrt(x^2+(16/(3*1^2))-4*x/1)
44 +sqrt(x^2+4*h^2*exp(-1*h)*(3*exp(-1*h)+1)/(3*(1-exp(-1*h))^2)
45 -4*x*h*exp(-1*h)/(1-exp(-1*h)));
46 zetah=-rhoh/(cte*(1-alphah)*gammah);
47
48 psi_min=(1/(1-alphah))*(betah+(rhoh/gammah)*(-1+log(zetah)));
49 hold on
50 plot(S,(psi_min)*ones(1,length(S)),'g')
51 end

```

# BIBLIOGRAPHIE

- [BDL16] Lucian Beznea, Madalina Deaconu, and Oana Lupaşcu. Stochastic equation of fragmentation and branching processes related to avalanches. *Journal of Statistical Physics*, 162(4) :824–841, Feb 2016.
- [Ber96] Jean Bertoin. *Lévy Processes*, volume 121 of *Cambridge Tracts in Mathematics*. Cambridge University Press, 1996.
- [Ber06] Jean Bertoin. *Random fragmentation and coagulation processes*, volume 102 of *Cambridge Studies in Advanced Mathematics*. Cambridge University Press, Cambridge, 2006.
- [BL16] Michel Benaïm and Claude Lobry. Lotka–volterra with randomly fluctuating environments or “how switching between beneficial environments can make survival harder”. *Ann. Appl. Probab.*, 26(6) :3754–3785, 12 2016.
- [BR11a] Nicole Bäuerle and Ulrich Rieder. *Markov decision processes with applications to finance*. Universitext. Springer, Heidelberg, 2011.
- [BR11b] Evelyn Buckwar and Martin G. Riedler. An exact stochastic hybrid model of excitable membranes including spatio-temporal evolution. *J. Math. Biol.*, 63(6) :1051–1093, 2011.
- [Bre81] Pierre Bremaud. *Point Processes and Queues : Martingale Dynamics*. Springer Series in Statistics. Springer New York, 1981.
- [BS78] Dimitri P. Bertsekas and Steven E. Shreve. *Stochastic Optimal Control : The Discrete-Time Case*. Academic Press, 1978.
- [CCF16] Fabien Campillo, Nicolas Champagnat, and Coralie Fritsch. Links between deterministic and stochastic approaches for invasion in growth-fragmentation-death models. *J. Math. Biol.*, 73(6-7) :1781–1821, 2016.

- [CD88] Oswaldo L.V. Costa and Mark H. A. Davis. Approximation for optimal stopping for a piecewise-deterministic process. *Mathematics of control, signals and systems*, 1(2) :123–146, 1988.
- [CDG<sup>+</sup>17] Bertrand Cloez, Renaud Dessalles, Alexandre Genadot, Florent Malrieu, Aline Marguet, and Romain Yvinec. Probabilistic and piecewise deterministic models in biology. *ESAIM : Procs*, 60 :225–245, 2017.
- [Che92] Mu-Fa Chen. *From Markov Chains to Non-Equilibrium Particle Systems*. WORLD SCIENTIFIC, 1992.
- [Clo11] Bertrand Cloez. Limit theorems for some branching measure-valued processes. *Advances in Applied Probability*, 49(2) :549–580, 2011.
- [CM16] Djalil Chafaï and Florent Malrieu. *Recueil de modèles aléatoires*, volume 78 of *Mathématiques et applications*. Springer, 2016.
- [Cos16] Manon Costa. A piecewise deterministic model for a prey-predator community. *Ann. Appl. Probab.*, 26(6) :3491–3530, 2016.
- [Dav84] Mark H. A. Davis. Piecewise-deterministic Markov processes : a general class of nondiffusion stochastic models. *J. Roy. Statist. Soc. Ser. B*, 46(3) :353–388, 1984. With discussion.
- [Dav93] Mark H. A. Davis. *Markov models and optimization*, volume 49 of *Monographs on Statistics and Applied Probability*. Chapman & Hall, London, 1993.
- [Dev86] Luc Devroye. *Non-Uniform Random Variate Generation*. Springer-Verlag, 1986.
- [DGR02] Vincent Dumas, Fabrice Guillemin, and Philippe Robert. A markovian analysis of additive-increase multiplicative-decrease algorithms. *Advances in Applied Probability*, 34(1) :85–111, 2002.
- [DHKR15] Marie Doumic, Marc Hoffmann, Nathalie Krell, and Lydia Robert. Statistical estimation of a growth-fragmentation model observed on a genealogical tree. *Bernoulli*, 21(3) :1760–1799, 2015.
- [dSCJ18] Benoîte de Saporta, Bertrand Cloez, and Maud Joubaud. Optimal stopping for measure-valued piecewise deterministic Markov processes. working paper or preprint, September 2018.
- [dSDG10] Benoîte de Saporta, François Dufour, and Karen Gonzalez. Numerical method for optimal stopping of piecewise deterministic Markov processes. *Ann. Appl. Probab.*, 20(5) :1607–1637, 2010. arXiv :0903.2114, hal-00367964.
- [dSDZ15] Benoîte de Saporta, François Dufour, and Huilong Zhang. *Numerical methods for simulation and optimization of piecewise deterministic Markov processes : application to reliability*. Mathematics and statistics series. Wiley-ISTE, 2015. hal-01249897.

- [FM04] Nicolas Fournier and Sylvie Méléard. A microscopic probabilistic description of a locally regulated population and macroscopic approximations. *Ann. Appl. Probab.*, 14(4) :1880–1919, 11 2004.
- [Fri14] Coralie Fritsch. *Approches probabilistes et numériques de modèles individus-centrés du chemostat*. PhD thesis, Montpellier, 2014.
- [Gat92] Dariusz Gatarek. Optimality conditions for impulsive control of piecewise-deterministic processes. *Mathematics of control, signals and systems*, 5(2) :217–232, 1992.
- [Gil76] Daniel T Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *Journal of Computational Physics*, 22(4) :403 – 434, 1976.
- [God03] Roger Godement. *Analyse mathématique IV*. Springer-Verlag Berlin Heidelberg, 2003.
- [GT14] Alexandre Genadot and Michèle Thiullen. Multiscale piecewise deterministic Markov process in infinite dimension : central limit theorem and Langevin approximation. *ESAIM Probab. Stat.*, 18 :541–569, 2014.
- [Gug86] Ulrich S. Gugerli. Optimal stopping of a piecewise-deterministic Markov process. *Stochastics*, 19(4) :221–236, 1986.
- [Guy07] Julien Guyon. Limit theorems for bifurcating Markov chains. Application to the detection of cellular aging. *Ann. Appl. Probab.*, 17(5-6) :1538–1569, 2007.
- [Kal86] Olav Kallenberg. *Random measures*. Academic Press, Inc., London, 1986.
- [Kol11] Vassili N. Kolokoltsov. *Markov processes, semigroups and generators*, volume 38 of *De Gruyter Studies in Mathematics*. Walter de Gruyter & Co., Berlin, 2011.
- [KP92] Peter E. Kloeden and Eckhard Platen. *Numerical Solution of Stochastic Differential Equations*, volume 23 of *Applications of Mathematics*. Springer, Berlin, Heidelberg, 1992.
- [LM84] J-P. Lepeltier and B. Marchal. Théorie générale du contrôle impulsif markovien. *SIAM Journal on Control and Optimization*, 22(4) :645–665, 1984.
- [Ouv98] Jean-Yves Oувrard. *Probabilités 1*. Cassini, 1998.
- [Ouv09] Jean-Yves Oувrard. *Probabilités 2*. Cassini, 3 edition, 2009.
- [PPP11] Gilles Pagès, Huyèn Pham, and Jacques Printems. An optimal markovian quantization algorithm for multidimensional stochastic control problems. *Stochastics and Dynamics*, 04, 11 2011.
- [RHK<sup>+</sup>14] Lydia Robert, Marc Hoffmann, Nathalie Krell, Stéphane Aymerich, Jérôme Robert, and Marie Doumic. Division in *escherichia coli* triggered by a size-sensing rather than a timing mechanism. *BMC Biology*, 12(1) :17, Feb 2014.



- [RTW12] Martin G. Riedler, Michèle Thieullen, and Gilles Wainrib. Limit theorems for infinite-dimensional piecewise deterministic Markov processes. Applications to stochastic excitable membrane models. *Electron. J. Probab.*, 17 :no. 55, 48, 2012.
- [Sha88] Michael Sharpe. *General Theory of Markov Processes*, volume 133 of *Pure and Applied Mathematics*. Academic Press, 1988.
- [Vil08] Cédric Villani. *Topics in Optimal Transportation*. Springer, 2008.
- [Vil09] Cédric Villani. *Optimal transport*, volume 338. Springer-Verlag Berlin Heidelberg, 2009.

# NOTATIONS

$B_b(W)$ ensemble de fonctions à valeurs réelles mesurables bornées sur $W$ .....	16
$(Z_n)_n$ chaîne incluse du PDMP réel .....	13
$(X_t)_t$ PDMP à valeurs mesure .....	19
$(X_t)_t$ PDMP réel .....	12
$\mathcal{B}(W), \mathcal{B}_b(W)$ resp. la tribu borélienne d'un espace $W$ et les boréliens bornés .....	9
$\mathcal{E}_x(l \circ \phi)$ loi " exponentielle " d'intensité $l \circ \phi$ (de fonction répartition $F$ ) .....	11
$E$ espace d'état du PDMP réel .....	9
$\mathcal{E}$ tribu de $E$ .....	9
$\phi, l, Q$ éléments caractéristiques du PDMP réel (flot, intensité, noyau) .....	10
$\mathcal{L}(X), \mathcal{L}(X \cdot)$ désigne la loi d'une variable aléatoire $X$ (resp. la loi conditionnelle selon $\cdot$ ) .....	12
$\mathcal{M}, \mathcal{M}_N$ ensemble des temps d'arrêt pour $(\mathcal{F}_t)_{t \geq 0}$ , et le sous-ensemble de ceux bornés par $T_N$ .....	33
$\mathcal{U}[0, 1]$ loi uniforme sur $[0, 1]$ .....	12
$I_p$ matrice identité de taille $p \times p$ .....	59
$\mathfrak{M}, \mathfrak{N}$ ensemble des mesures localement finies sur $\mathbb{R}^m$ , et sous-ensemble des mesures discrètes .....	16
$\partial W, \overline{W}$ désigne resp. le bord d'un espace $W$ et sa fermeture .....	9
$\mathcal{P}(H)$ ensemble des mesures de probabilités sur $H$ .....	59
$\rho$ distance sur l'espace $E$ .....	10

$\mathcal{W}_p$	distance de Wasserstein d'ordre $p$ ( $\mathcal{W}_1$ étant juste notée $\mathcal{W}$ ) .....	76
$C_c(W)$	ensemble de fonctions à valeurs réelles continues à support compact sur $W$ .....	16
$D_E[0, +\infty[$	ensemble des fonctions <i>càdlàg</i> sur $\mathbb{R}_+$ à valeurs dans $E$ .....	15
$K$	ensemble des modes du PDMP réel .....	9
$X \perp\!\!\!\perp Y$	indépendance entre deux v.a. $X$ et $Y$ .....	59



## Résumé.

Les processus markoviens déterministes par morceaux (PDMP) forment une vaste classe de processus stochastiques caractérisés par une évolution déterministe entre des sauts à mécanisme aléatoire. Ce sont des processus de type hybride, avec une composante discrète de mode et une composante d'état qui évolue dans un espace continu. Entre les sauts du processus, la composante continue évolue de façon déterministe, puis au moment du saut un noyau markovien sélectionne la nouvelle valeur des composantes discrète et continue. Dans cette thèse, nous construisons des PDMP évoluant dans des espaces de mesures (de dimension infinie), pour modéliser des population de cellules en tenant compte des caractéristiques individuelles de chaque cellule. Nous exposons notre construction des PDMP sur des espaces de mesure, et nous établissons leur caractère markovien. Sur ces processus à valeur mesure, nous étudions un problème d'arrêt optimal. Un problème d'arrêt optimal revient à choisir le meilleur temps d'arrêt pour optimiser l'espérance d'une certaine fonctionnelle de notre processus, ce qu'on appelle fonction valeur. On montre que cette fonction valeur est solution des équations de programmation dynamique et on construit une famille de temps d'arrêt  $\varepsilon$ -optimaux. Dans un second temps, nous nous intéressons à un PDMP en dimension finie, le TCP, pour lequel on construit un schéma d'Euler afin de l'approcher. Ce choix de modèle simple permet d'estimer différents types d'erreurs. Nous présentons des simulations numériques illustrants les résultats obtenus.

**Mots clés :** *Processus de Markov déterministe par morceaux, contrôle stochastique, simulation, optimisation*

## Abstract.

Piecewise deterministic Markov processes (PDMP) form a large class of stochastic processes characterized by a deterministic evolution between random jumps. They fall into the class of hybrid processes with a discrete mode and an Euclidean component (called the state variable). Between the jumps, the continuous component evolves deterministically, then a jump occurs and a Markov kernel selects the new value of the discrete and continuous components. In this thesis, we extend the construction of PDMPs to state variables taking values in some measure spaces with infinite dimension. The aim is to model cells populations keeping track of the information about each cell. We study our measured-valued PDMP and we show their Markov property. With those processes, we study an optimal stopping problem. The goal of an optimal stopping problem is to find the best admissible stopping time in order to optimize some function of our process. We show that the value function can be recursively constructed using dynamic programming equations. We construct some  $\varepsilon$ -optimal stopping times for our optimal stopping problem. Then, we study a simple finite-dimension real-valued PDMP, the TCP process. We use Euler scheme to approximate it, and we estimate some types of errors. We illustrate the results with numerical simulations.

**Key words :** *Piecewise deterministic Markov process, stochastic control, simulation, optimization*