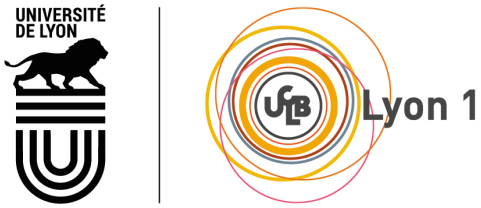


N° d'ordre NNT : 2018LYSE1155

THÈSE DE DOCTORAT DE L'UNIVERSITÉ DE LYON
opérée au sein de
l'Université Claude Bernard Lyon 1

École Doctorale ED512
InfoMaths

Spécialité de doctorat : Mathématiques

Soutenue publiquement le 27/09/2018, par :
Ulysse Herbach

Modélisation stochastique de l'expression des gènes et inférence de réseaux de régulation

---

Devant le jury composé de :

Pascale Crépieux, DR, CNRS	Rapporteure
Nadine Guillotin-Plantard, MCU, Université Lyon 1	Examinatrice
Eva Löcherbach, PR, Université de Cergy-Pontoise	Examinatrice
Christian Mazza, PR, Université de Fribourg	Rapporteur
Benoîte de Saporta, PR, Université de Montpellier	Présidente du jury
Romain Yvinec, CR, INRA	Examinateur
Thibault Espinasse, MCU, Université Lyon 1	Directeur de thèse
Anne-Laure Fougères, PR, Université Lyon 1	Directrice de thèse (invitée)
Olivier Gandrillon, DR, CNRS	Directeur de thèse

UNIVERSITE CLAUDE BERNARD - LYON 1

Président de l'Université	**M. le Professeur Frédéric FLEURY**
Président du Conseil Académique	M. le Professeur Hamda BEN HADID
Vice-président du Conseil d'Administration	M. le Professeur Didier REVEL
Vice-président du Conseil Formation et Vie Universitaire	M. le Professeur Philippe CHEVALIER
Vice-président de la Commission Recherche	M. Fabrice VALLÉE
Directrice Générale des Services	Mme Dominique MARCHAND

COMPOSANTES SANTE

Faculté de Médecine Lyon Est – Claude Bernard	Directeur : M. le Professeur G.RODE
Faculté de Médecine et de Maïeutique Lyon Sud – Charles Mérieux	Directeur : Mme la Professeure C. BURILLON
Faculté d'Odontologie	Directeur : M. le Professeur D. BOURGEOIS
Institut des Sciences Pharmaceutiques et Biologiques	Directeur : Mme la Professeure C. VINCIGUERRA
Institut des Sciences et Techniques de la Réadaptation	Directeur : M. X. PERROT
Département de formation et Centre de Recherche en Biologie Humaine	Directeur : Mme la Professeure A-M. SCHOTT

COMPOSANTES ET DEPARTEMENTS DE SCIENCES ET TECHNOLOGIE

Faculté des Sciences et Technologies	Directeur : M. F. DE MARCHI
Département Biologie	Directeur : M. le Professeur F. THEVENARD
Département Chimie Biochimie	Directeur : Mme C. FELIX
Département GEP	Directeur : M. Hassan HAMMOURI
Département Informatique	Directeur : M. le Professeur S. AKKOUCHE
Département Mathématiques	Directeur : M. le Professeur G. TOMANOV
Département Mécanique	Directeur : M. le Professeur H. BEN HADID
Département Physique	Directeur : M. le Professeur J-C PLENET
UFR Sciences et Techniques des Activités Physiques et Sportives	Directeur : M. Y.VANPOULLE
Observatoire des Sciences de l'Univers de Lyon	Directeur : M. B. GUIDERDONI
Polytech Lyon	Directeur : M. le Professeur E.PERRIN
Ecole Supérieure de Chimie Physique Electronique	Directeur : M. G. PIGNAULT
Institut Universitaire de Technologie de Lyon 1	Directeur : M. le Professeur C. VITON
Ecole Supérieure du Professorat et de l'Education	Directeur : M. le Professeur A. MOUGNIOTTE
Institut de Science Financière et d'Assurances	Directeur : M. N. LEBOISNE

À Michel,
qui m'a transmis cet enthousiasme.

Remerciements

Mes premiers remerciements vont tout naturellement à mes directeurs/trice de thèse, Olivier Gandrillon, Thibault Espinasse et Anne-Laure Fougères. Sacré triptyque, car vous avez tous les trois joué un rôle bien différent mais complémentaire, et je ne pouvais sincèrement pas rêver mieux pour me guider à travers cette aventure qu'est la thèse. Grâce à vous, j'ai passé trois années palpitantes. Olivier, j'espère que ma thèse aura eu la bonne granularité... Thibault, n'oublie pas de faire homologuer ton record de lancer de pistes à la minute ; Anne-Laure, merci de m'avoir évité de tomber dans un min local !

Je suis par ailleurs extrêmement reconnaissant à Pascale Crépieux et Christian Mazza d'avoir rapporté ma thèse, ainsi qu'à Nadine Guillotin-Plantard, Eva Löcherbach, Benoîte de Saporta et Romain Yvinec d'avoir accepté de faire partie de mon jury. En outre, je remercie chaleureusement Florent Malrieu, Dan Goreac et Simon Masnou pour leur participation à mon comité de suivi de thèse.

Sur le campus de la Doua, viennent ensuite les anciens professeurs du Master *maths en action* que j'ai eu la chance de recroiser durant ces années de thèse : Franck Picard (les conseils éclairés), Laurent Jacob (le poly dont j'ai malencontreusement déchiré la première page) et Vivian Viallon (les bonnes questions qui fâchent). Dans un autre registre je pense aussi à Sandrine Charles (je me demande si les saumons mangent des daphnies) et Philippe Veber (les bonnes questions qui ne fâchent pas). Finalement, je tiens à remercier les équipes DRACULA et BM2A au grand complet, qui m'ont accueilli dès le stage de Master.

Les jeunes ne sont bien sûr pas en reste : Loïs, Aurélien et Simon d'un coté, Angélique, Arnaud (allez ça passe), Anissa et Ronan de l'autre, et enfin les doctorant·e·s du comité d'Inter'Actions 2018, avec qui ce fut un plaisir de n'effectuer qu'un malheureux dixième de leur considérable travail d'organisation.

L'équipe BM2A a droit à un deuxième remerciement puisqu'elle a déménagé et s'appelle désormais SBDM, et aussi car ce fut, grâce à son excellente ambiance, un cadre particulièrement agréable et stimulant pour travailler à l'interface entre mathématiques et biologie. Angélique, merci pour toutes tes explications qui m'ont mis le pied à l'étrier, et Geneviève, merci pour tes cours de chromatine et ta veille scientifique assidue...

À ce stade il y a encore bien des personnes que j'aimerais remercier : si ces dernières venaient à lire ces lignes, elles ne se formaliseraient pas de leur absence desdites lignes car elles comprendraient que ce manuscrit devait vite partir à l'impression... Ouf !

Enfin, *last but not least*, ma chère Latifa B : `SRFKIM`.

Table des matières

Introduction

Intelligence is the ability to adapt to change. Stephen Hawking [1]

Comment une cellule fait-elle pour prendre des décisions ? La question peut paraître assez saugrenue si l'on imagine cette cellule comme le simple exécutant d'un *programme génétique* codé par la séquence de nucléotides – les fameuses lettres A, C, G, T – de son ADN. À l'époque du *central dogma of molecular biology*, hypothèse scientifique [2] formulée par Francis Crick en 1958 et qui est toujours en accord avec l'expérience, ce point de vue est tout à fait cohérent : selon cette hypothèse, l'information nécessaire au fonctionnement de la cellule se transmet physiquement de l'ADN vers l'ARN messager (ARNm), puis de l'ARNm vers les protéines, dont il existe un très grand nombre de types et qui sont les molécules de base du milieu cellulaire. On parle de *transcription* pour la première étape (l'alphabet est le même à une lettre près) et de *traduction* pour la deuxième (l'alphabet change complètement). L'information contenue dans l'ADN se propage donc dans un seul sens, et il est tentant de voir tout simplement la cellule comme un ordinateur qui interpréterait du code : une cellule n'a pas de décision à prendre... elle ne fait que lire le code ! On définit alors naturellement les *gènes* comme étant les portions de la séquence ADN qui codent effectivement pour des protéines.

Ce parallèle avec l'informatique est rassurant et fonctionne très bien jusqu'à un certain point. Mais en allant un peu plus loin dans la description de l'activité moléculaire au sein de la cellule, il s'avère que l'interprétation du code peut changer... en fonction des protéines ambiantes. Il y a donc bien de l'information qui passe des protéines vers l'ADN, sous forme de modification non pas de la séquence elle-même mais de la lecture de cette séquence. On parle alors de *l'expression* d'un gène pour désigner la quantité d'ARNm effectivement produit correspondant à ce gène. Deux cellules ayant les mêmes gènes n'ont a priori aucune raison d'exprimer ces gènes de la même façon : il se trouve que c'est justement l'un des mécanismes principaux par lesquels des cellules « souches » pluripotentes se différencient en cellules « matures » spécialisées (neurones, globules rouges, etc.). Nous arrivons bel et bien à la notion de prise de décision : pourquoi telle cellule se différencie alors que telle autre reste dans un état d'auto-renouvellement ?

Pour répondre à cette question, il faut commencer par considérer la cellule comme un système complexe – l'un des principaux objets d'étude de la *biologie des systèmes* – dont l'expression de chaque gène peut être régulée, typiquement par des protéines produites par d'autres gènes et baptisées pour l'occasion « facteurs de transcription ». Le principe d'une

1. www.washingtonpost.com/news/answer-sheet/wp/2018/03/29/stephen-hawking-famously-said-intelligence-is-the-ability-to-adapt-to-change-but-did-he-really-say-it

2. Jacques Monod lui expliquera quelques années plus tard qu'il aurait mieux fait de regarder dans un dictionnaire avant d'utiliser le mot *dogma*.

Figure 1 – Diagramme fondamental de la biologie moléculaire. Le *central dogma* à proprement parler rajoute d'autres flèches, mais jamais des protéines vers l'ADN : notre point de départ consistera justement à rajouter cette flèche pour représenter la régulation de l'expression des gènes.

hiérarchie restant rassurant, la notion de *master regulators* est devenue populaire, l'idée étant que la régulation est orchestrée par un petit nombre de gènes, eux-mêmes répondant de manière stéréotypée à des stimuli extérieurs. Mais il s'est avéré que ces gènes spéciaux pouvaient être eux-mêmes régulés... À tel point qu'il devient assez vain de chercher un chef d'orchestre. Aujourd'hui, on parle plutôt de *réseau de régulation* entre les gènes, et l'on peut même aller plus loin avec une autre hypothèse scientifique qui est le contexte de cette thèse : *le comportement de la cellule est une propriété émergente des interactions entre les gènes.*

Le « dogme central » ne parle donc que d'une partie de l'iceberg puisqu'il met de coté l'influence des protéines sur la lecture du génome (Figure 1). Or comme tout iceberg digne de ce nom, celui-ci possède une partie immergée assez conséquente. Plus précisément, la fameuse séquence de nucléotides de l'ADN est rarement présente seule mais plutôt entourée par un nombre gigantesque d'autres molécules (typiquement encore des protéines, par exemple les histones autour desquels la séquence s'enroule), l'ensemble constituant la *chromatine* chez les eucaryotes. Cette forme permet notamment à la séquence d'être compactée, le niveau le plus élevé étant celui de chromosome. Pour avoir une idée de l'ampleur du phénomène, Liu et Tjian (2018) nous aident avec des chiffres :

> *Roughly 2m (approximately six billion base pairs) of linear DNA must be packed into the nucleus of each diploid human cell (∼5–10μm in diameter). With the same packaging density, it is possible to put a strand of DNA >6,000 times the Earth's circumference inside a chicken egg.*

C'est impressionnant. Mais surtout, il y a un autre aspect fondamental : la régulation de l'expression est en lien étroit avec les changements d'état de la chromatine (Cremer *et al.*, 2015 ; Bintu *et al.*, 2016). En effet, en tant que réaction chimique nécessitant un contact avec des molécules ambiantes cruciales, la transcription a plus de chances de se produire aux endroits où la chromatine est plutôt « ouverte » (décompactée), et beaucoup moins aux endroits où la chromatine est plutôt « fermée » (compactée). Or on sait maintenant que l'état de la chromatine peut se transmettre lors de la division (Hathaway *et al.*, 2012 ; Kueng *et al.*, 2013). Il y a donc une forme d'hérédité des motifs d'expression de gènes qui n'est pas liée à des changements de la séquence ADN... on parle alors de caractères *épigénétiques*.

Il est intéressant de constater que le terme *epigenetic* avait été introduit par Conrad Waddington dès 1942, alors qu'à l'époque la nature physique des gènes n'était pas encore bien connue. Programme génétique ou pas, on avait bien remarqué qu'un organisme multicellulaire se forme par divisions successives d'une cellule de départ, et que les cellules finissent par se différencier les unes des autres. En outre, il est possible d'étudier la différenciation sans considérer l'ADN explicitement : on représente les cellules comme des particules évoluant dans l'espace des niveaux d'expression des gènes – de très grande dimension – avec un certain potentiel énergétique, ce qui aboutit à l'idée de *paysage épigénétique*. Cette idée a été sérieusement remise au goût du jour (Huang, 2009 ; Davila-Velderrain *et al.*, 2015 ;

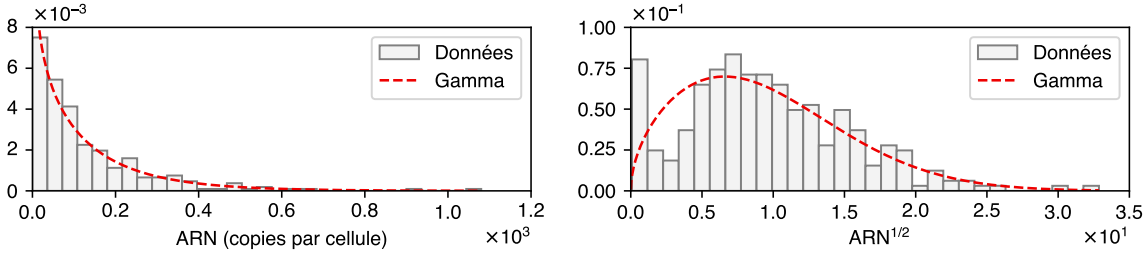


Figure 2 – Allure typique de la distribution de l'expression d'un gène au niveau *single-cell*. Les données exprimées en nombre de molécules semblent bien correspondre à une loi Gamma (gauche). Si l'on comprime les grandes valeurs en appliquant la transformation racine carrée, la distribution devient bimodale et la loi Gamma est incapable de reproduire ce motif (droite).

Moris *et al.*, 2016). Les types cellulaires sont alors bien moins figés et correspondent plutôt à des bassins d'attraction dans l'espace d'états, le choix de différenciation d'une cellule correspondant au passage d'un bassin d'attraction à un autre (Huang, 2010).

C'est le point de vue que nous adoptons dans cette thèse et ce n'est pas vraiment un hasard : en fait, c'est justement parce qu'il accorde une place au hasard. L'expression des gènes a longtemps été observable uniquement à travers des quantités moyennes mesurées sur des populations de cellules : l'évolution moyenne étant tout à fait reproductible, cela laissait penser que ces dernières se comportaient toutes de la même façon, répondant de manière déterministe à des stimuli. L'arrivée des techniques *single-cell*, en plein essor, permet aujourd'hui d'observer des niveaux d'ARNm et de protéines dans des cellules individuelles. Il s'avère que même au sein d'une population de cellules de génome identique et dans un même environnement, la variabilité entre les cellules à l'échelle moléculaire est parfois très forte (Figure 2), au point qu'une description moyenne est clairement insuffisante pour étudier certains phénomènes biologiques comme la différenciation.

Stochasticité de l'expression des gènes

Après avoir été considérée pendant un certain temps comme un « bruit » pour les cellules, cette variabilité de l'expression s'est bien établie dans la littérature (Kaern *et al.*, 2005) et il est maintenant clair qu'elle joue un rôle majeur et non-nécessairement néfaste (Huang, 2009 ; Eldar et Elowitz, 2010 ; Chalancon *et al.*, 2012). Certains vont même plus loin :

> *A large amount of data demonstrating the stochastic nature of gene expression and cell differentiation has accumulated during the last 40 years. These data suggest that a gene in a cell always has a certain probability of being activated at any time and that instead of leading to on and off switches in an all-or-nothing fashion, the concentration of transcriptional regulators increases or decreases this probability.* (Laforge *et al.*, 2005)

Adopter une perspective fondamentalement probabiliste de l'expression des gènes pourrait donc se révéler très fructueux. On ne pourra pas manquer de faire un lien avec la physique statistique,[3] et d'ailleurs des physiciens travaillent précisément sur le sujet (Friedman *et al.*, 2006 ; Kim et Wang, 2007). Comme le disent encore Laforge *et al.* (2005), il est bien connu

3. Le sens du mot *statistique* a quelque peu changé : si la physique statistique devait être inventée de nos jours, elle s'appellerait certainement plutôt « physique probabiliste ».

dans cette discipline que des phénomènes aléatoires au niveau moléculaire peuvent conduire à des structures macroscopiques organisées : il n'y a donc pas de paradoxe entre la stochasticité de l'expression à l'échelle d'une cellule et le fait que le développement d'un organisme soit un phénomène tout à fait reproductible.

Dans toute la suite, on se placera ainsi à l'échelle d'une cellule dont on décrira les gènes comme un système de particules en interaction, fournissant les « coordonnées » de la cellule vue à son tour comme particule dans l'espace des gènes. L'avantage de l'approche statistique est d'apporter une notion rigoureuse de potentiel énergétique que l'on interprètera comme une version contemporaine du paysage de Waddington (Li et Wang, 2013), si bien que l'on pourrait parler de *biologie statistique*. Au moins une différence avec la physique statistique classique : les gènes ne sont pas des particules indiscernables (en particulier on ne cherchera pas une traditionnelle « propagation du chaos »), et d'ailleurs on oscillera entre ce point de vue et celui d'un *many-body problem*, idée introduite de façon remarquable par Sasai et Wolynes (2003) dont nous reparlerons. En revanche, certaines propriétés *statistiques* de ce système de particules sont très prometteuses dans l'étude de la décision cellulaire au niveau macroscopique : c'est le cas de l'entropie, dont nous reparlerons aussi.

Biologie des systèmes et inférence de réseaux

De manière plus pragmatique, on s'intéresse dans cette thèse à un problème classique en biologie des systèmes : l'inférence de réseaux de régulation. En effet, puisque ces réseaux sont à la base du comportement des cellules, on voudrait pouvoir les reconstruire à partir de données faciles à acquérir pour éviter de tester précisément chaque interaction une par une. Et même s'il s'avérait impossible de reconstruire parfaitement toutes les interactions, on aimerait au moins disposer d'un outil statistique pour guider l'expérience, par exemple en révélant des motifs particuliers dans le réseau, ou en faisant des prédictions sur le comportement de la cellule si l'on agit sur tel ou tel gène.

Les données les plus accessibles à l'heure actuelle sont les niveaux d'ARNm, aussi appelés *mesures d'expression* ou *données transcriptomiques*. Comme évoqué plus haut, celles-ci se divisent globalement en deux types : les données de *population* et les données de *cellules uniques*. C'est bien sûr le deuxième type qui contient l'information la plus riche : pour un instant donné, on dispose des mesures simultanées des niveaux d'expression d'un ensemble de gènes, et ce pour un certain nombre de cellules individuelles. D'un point de vue statistique, on a donc accès à la *loi jointe*.[4] En comparaison, les données de population correspondent simplement à la moyenne de chaque gène. Détail qui a son importance, les techniques de mesure impliquent pour la plupart la mort des cellules : lorsqu'on effectue des mesures à plusieurs instants, par exemple pour étudier un retour à l'équilibre après une perturbation, on n'a donc pas les trajectoires de chaque cellule mais plutôt des *snapshots*, c'est-à-dire des échantillons indépendants d'une distribution qui évolue dans le temps.

Concernant l'inférence à proprement parler, le moins que l'on puisse dire c'est que le terrain est déjà bien arpenté. Ces données sont tellement accessibles que l'enjeu pour les biologistes est surtout de trouver des outils pertinents pour les analyser. Du pain béni pour les statisticiens ! À peu près toutes les méthodes multivariées en circulation ont été appliquées à l'analyse de données d'expression, et en particulier à l'inférence de réseaux de gènes (Hecker *et al.*, 2009 ; Wang et Huang, 2014). À titre d'exemple, l'un des algorithmes d'inférence les

4. On utilisera de manière interchangeable *loi* et *distribution* pour désigner des mesures de probabilité.

plus populaires est basé sur les forêts aléatoires (Huynh-Thu *et al.*, 2010) tandis que l'outil incontournable en première phase d'analyse reste la traditionnelle analyse en composantes principales (ACP). Il faut noter que la grande majorité des techniques existantes est basée sur les données de population : en particulier le cadre probabiliste est souvent imposé a priori pour sa simplicité (Gallopin *et al.*, 2013) voire pas complètement posé (Ghazanfar *et al.*, 2016). Le coté *single-cell* est plus récent mais souffre un peu du même problème, avec principalement des approches heuristiques (Babtie *et al.*, 2017).

Il y a globalement un obstacle majeur, souligné par exemple par Chan *et al.* (2017) et que l'on pourrait presque appeler le « fléau de la biologie des systèmes » : il est très facile de produire des graphes (orientés ou non) à partir des données, mais très difficile de savoir si ces graphes correspondent à une réalité biologique. Le dilemme est le suivant : ou bien l'on simule des données de test avec un modèle (*in silico*) et alors on est certain du réseau mais sans garantie que les données soient réalistes (et en pratique c'est encore loin d'être le cas à l'échelle de la cellule unique), ou bien l'on prend des données réelles (*in vitro*) mais alors on n'est jamais certain du réseau, sachant que dans ce domaine les précieux *gold standards* sont généralement eux-mêmes issus d'une inférence statistique plus ancienne...

C'est dans ce contexte que l'on propose de poser la question autrement : peut-on envisager une approche « mécaniste » pour l'inférence de réseaux de régulation, c'est-à-dire directement reliée à un modèle biochimique à l'échelle de la cellule ? Cette idée non plus n'est pas vraiment nouvelle puisque divers modèles dynamiques à base d'équations différentielles ont déjà été utilisés sur des données de population, décrivant ainsi des trajectoires moyennes dans l'espace des gènes (Mizeranschi *et al.*, 2015). Cependant, comme il n'existe pas d'équation prétendant décrire la moyenne qui fasse consensus sur le plan de la description physique, les modèles déterministes restent assez empiriques. Et surtout, l'arrivée des données de cellules uniques change deux choses :

- l'accès aux *dépendances statistiques* entre les gènes, qui n'étaient accessibles jusque là que de manière indirecte en perturbant artificiellement l'expression, offre – si celles-ci se révèlent non triviales – un gain de précision énorme dans l'observation des réseaux ;
- le fait d'avoir accès à la distribution, et non plus seulement sa moyenne, permet de « descendre » dans l'échelle de la modélisation et en particulier d'utiliser une description *linéaire* au niveau microscopique, même si le comportement devient *non linéaire* au niveau macroscopique.

Tout ceci motive fortement une approche mécaniste. On a vu que l'expression des gènes était fondamentalement probabiliste : il s'agit donc de construire un modèle stochastique de réseau de gènes, basé sur des interactions de type activation/inhibition, qui puisse à la fois décrire les observations – à partir d'arguments physiques plutôt qu'empiriques – et fournir une méthode statistique utilisable en pratique sur des données réelles.[5]

Il nous reste à préciser ce que l'on entend par « réseau » car le sens n'est pas toujours clair dans la littérature. Nous ferons bien la distinction entre les interactions fondamentales des gènes, qui proviennent de leur structure physique et que nous supposerons fixées dans le temps (hypothèse qui semble réaliste à notre échelle d'observation), et d'autre part les dépendances statistiques des niveaux d'expression : dans tout ce qui suit, un *réseau* sera un ensemble donné d'interactions – le réseau a donc une *structure* fixe représentée par un graphe orienté – tandis que la distribution jointe des niveaux d'expression à un instant

5. On sent qu'il va falloir trouver un compromis entre ces deux directions. D'un point de vue pratique, l'idéal est simplement d'exploiter au mieux l'information potentiellement contenue dans les données *single-cell*.

donné correspondra à un *état* du réseau et pourra varier dans le temps. Il semblerait que la possibilité de confusion vienne du fait que les graphes obtenus à partir des dépendances statistiques sont très souvent appelés « réseaux de corrélation » : ceux-là peuvent clairement varier dans le temps.

Un exemple typique est l'activation d'une « voie de signalisation » : un gène qui n'était auparavant pas transcrit voit son niveau d'expression augmenter, puis l'ARNm est traduit en protéines qui vont à leur tour déclencher la transcription d'un autre gène, etc. Pour nous, cette voie est toujours présente dans le réseau de régulation, mais elle n'est simplement pas détectable avant que le premier gène n'atteigne un certain niveau d'expression. Du point de vue de l'inférence, il est clair que l'on ne sera capable de reconstruire cette voie que si les observations correspondent à une période où elle est utilisée : pour faire une analogie assez simpliste, on ne peut pas savoir si les ampoules sont branchées lorsqu'on a une panne de courant. Au final, le but n'est pas d'avoir le réseau « complet », qui correspondrait à une connaissance exhaustive du fonctionnement de la cellule, mais plutôt de reconstruire des parties de ce réseau – que l'on pourra bien sûr appeler aussi des réseaux, tout est relatif – associées à un certain phénomène étudié, par exemple la différenciation.

Vue d'ensemble de la thèse

L'objectif de cette thèse est de développer un cadre et des résultats mathématiques à la fois rigoureux et directement applicables au contexte biologique. Pour permettre une navigation plus facile, nous avons opté pour une organisation globale en deux parties, la première étant résolument plus « mathématique » et la deuxième plus « biologique ». Cela dit, la première partie s'attachera à mettre en évidence l'interprétation biologique et la deuxième partie contiendra aussi des maths. Notre espoir est que chaque discipline y trouve son compte, sachant que nos recherches nous ont fait entrevoir, dans la régulation de l'expression des gènes, aussi bien un sujet de biologie passionnant qu'un domaine d'application des mathématiques encore très peu exploré par rapport à sa richesse potentielle.

Cette thèse s'articule plus précisément en sept chapitres, les trois premiers étant consacrés à notre approche mathématique et les quatre suivants aux données biologiques et à la mise en pratique des résultats.

- Le chapitre 1 détaille la construction d'un modèle stochastique de réseau de gènes, sous la forme d'un processus de Markov déterministe par morceaux (PDMP). Le point de départ est le premier modèle stochastique non trivial d'expression d'un gène, le *modèle à deux états*. Le modèle de réseau final est notre « gold standard » : c'est fondamentalement un modèle physique (i.e. mécaniste). Nous construisons notamment un sous-modèle de chromatine – lui aussi stochastique mais bien plus simple – afin de justifier la forme des interactions entre les gènes.
- Le chapitre 2 est au cœur de l'approche mécaniste : il s'agit d'obtenir une approximation sous forme explicite de la loi stationnaire du PDMP, puis d'interpréter celle-ci comme une vraisemblance statistique de façon à ce que l'inférence de réseau corresponde en fait au calibrage du modèle physique. Après une étape de simplification, nous décrivons deux stratégies différentes : la première correspond à une « approximation de champ » assez populaire en physique, pour laquelle nous obtenons un résultat de concentration, et la deuxième se base sur un cas particulier que l'on résout de manière exacte, ce qui aboutit à un modèle statistique bien posé sous la forme d'un champ de Markov caché.

- Le chapitre 3 décrit une généralisation naturelle du modèle à deux-états et correspond à un article soumis pour publication. Il est un peu à part et, même s'il y a bien une motivation biologique, c'est le chapitre le plus tourné vers les mathématiques. Nous obtenons dans ce cadre une forme explicite pour la loi stationnaire, tout en soulignant un lien algébrique étroit entre les modèles chimiques « exacts » basés sur des processus de sauts et leurs versions « hybrides » basées sur des PDMP.
- Le chapitre 4 est celui qui est le plus tourné vers la biologie. Il présente un article publié en 2016 dans *PLOS Biology*, autour d'un travail mené par Angélique Richard, qui met en lumière l'importance de la variabilité au cours de la différenciation grâce à une analyse au niveau *single-cell*. Les chapitres suivants sont notamment basés sur ces données.
- Le chapitre 5 est le premier exemple d'application du modèle de réseau : il correspond à un article soumis, issu d'un travail mené par Arnaud Bonnaffoux, qui présente une approche itérative utilisant simplement le modèle comme boite noire pour le comparer aux observations. Il s'agit d'une méthode *brute-force* mais très flexible, avec l'avantage de pouvoir exploiter directement l'aspect temporel des données.
- Le chapitre 6 présente un article publié en 2017 dans *BMC Systems Biology* et se base sur la première stratégie développée au chapitre 2, que l'on appelle ici *approximation de Hartree*. Le principe est de maximiser une pseudo-vraisemblance explicite, ce que nous faisons via une approche de type « hard EM ». Nous montrons que malgré son caractère approximatif la méthode permet déjà de retrouver des réseaux simples de 2 gènes.
- Le chapitre 7 détaille l'application du champ de Markov caché qui est suggéré par la deuxième stratégie du chapitre 2. On l'appelle ici *auto-modèle Gamma-Binomial* en raison de sa forme particulière. Il s'agit cette fois d'un cadre probabiliste bien posé et nous proposons une méthode d'inférence dite « variationnelle » qui exploite un résultat de convexité intéressant.

Pour finir, on trouvera un dernier chapitre de conclusion et perspectives. Avant de se lancer, il ne reste plus qu'à préciser quelques...

Notations et conventions

L'idée de notations uniformes sur toute la thèse s'étant révélée irréaliste, nous avons préféré assurer une cohérence « chapitre par chapitre ». Les notations utiles seront donc introduites dans chaque chapitre, ce qui devrait théoriquement éviter toute confusion. On peut quand même signaler qu'au niveau global nous noterons :

- $[\![1, n]\!]$ pour désigner l'ensemble $\{1, 2, \dots, n\}$ lorsque $n \in \mathbb{N}^*$;
- $]a, b[$ ou (a, b) pour les intervalles ouverts de $\mathbb{R}$ (notation anglo-saxonne) ;
- $\mathcal{M}_n(\mathbb{R})$ ou $\mathbb{R}^{n\times n}$ l'ensemble des matrices réelles carrées de taille n (idem) ;
- $\mathcal{L}(X)$ ou $\mathbb{P}_X$ pour désigner la loi (distribution) d'une variable aléatoire X ;
- $\dot{X}_t$ la dérivée en temps d'une fonction potentiellement aléatoire $t \mapsto X_t$;
- ∂_x l'opérateur de dérivation partielle par rapport à x ;
- Γ et B les fonctions Gamma et Beta définies pour $a, b \in \mathbb{R}_+^*$ par

$$\Gamma(a) = \int_0^\infty x^{a-1}e^{-x}dx \quad \text{et} \quad \mathrm{B}(a, b) = \int_0^1 x^{a-1}(1-x)^{b-1}dx = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}.$$

Variables aléatoires et processus. Toutes les variables aléatoires seront implicitement définies sur un même espace de probabilité, et les processus stochastiques considérés auront toujours une construction naturelle : ce sont tous des processus de Feller et on les définira simplement par leur générateur.

Espèces chimiques. Les modèles mathématiques que l'on construira seront basés sur le formalisme des systèmes de réactions chimiques. On notera A, B, C... les espèces chimiques considérées, tandis que $[\mathsf{A}]$ désignera la quantité de l'espèce A à un instant donné. Selon le choix de modélisation, on aura tantôt $[\mathsf{A}] \in \mathbb{N}$ (nombre de molécules de A), tantôt $[\mathsf{A}] \in \mathbb{R}_+$ (quantité continue de A, concentration chimique si divisée par un volume).

Produit tensoriel. On utilisera parfois la notion de produit tensoriel d'espaces vectoriels de dimension finie. Nous resterons à un niveau élémentaire et ce sera surtout une façon de simplifier les calculs. Étant donné un espace vectoriel E de dimension d, nous noterons $E^{\otimes n} = E \otimes \cdots \otimes E$ le produit tensoriel de n copies de E. Il s'agit d'un espace vectoriel de dimension d^n, caractérisé par le fait qu'il existe une application n-linéaire $\phi : E^n \to E^{\otimes n}$, définissant le produit tensoriel de vecteurs $u_1, \dots, u_n \in E$ comme étant l'élément

$$u_1 \otimes \cdots \otimes u_n = \phi(u_1, \dots, u_n),$$

telle que si $(e_1, \dots, e_d)$ est une base de E, alors $(e_{k_1} \otimes \cdots \otimes e_{k_n})_{1 \leqslant k_1, \dots, k_n \leqslant d}$ est une base de $E^{\otimes n}$. Ainsi, lorsque $u_i = \alpha_{i,1} e_1 + \cdots + \alpha_{i,d} e_d$ pour $i \in [\![1, n]\!]$, on a la relation fondamentale

$$\bigotimes_{i=1}^{n} u_i = u_1 \otimes \cdots \otimes u_n = \sum_{1 \leqslant k_1, \dots, k_n \leqslant d} \alpha_{k_1, \dots, k_n} e_{k_1} \otimes \cdots \otimes e_{k_n}$$

où $\alpha_{k_1, \dots, k_n} = \alpha_{1,k_1} \alpha_{2,k_2} \cdots \alpha_{n,k_n}$. Noter que les coordonnées $\alpha_{k_1, \dots, k_n}$ d'un élément quelconque de $E^{\otimes n}$ ne se factorisent pas nécessairement de la sorte. Soit maintenant F un autre espace vectoriel et $f_1, \dots, f_n$ des applications linéaires de E dans F. Le produit tensoriel $f_1 \otimes \cdots \otimes f_n$ est l'application linéaire de $E^{\otimes n}$ dans $F^{\otimes n}$ définie par :

$$(f_1 \otimes \cdots \otimes f_n)(u_1 \otimes \cdots \otimes u_n) = f_1(u_1) \otimes \cdots \otimes f_n(u_n), \quad \forall (u_1, \dots, u_n) \in E^n.$$

On utilisera notamment le résultat suivant, immédiat mais très utile : *si pour tout $i \in [\![1, n]\!]$, u_i est un vecteur propre de f_i associé à la valeur propre λ_i, alors $u_1 \otimes \cdots \otimes u_n$ est un vecteur propre de $f_1 \otimes \cdots \otimes f_n$ associé à la valeur propre $\lambda_1 \cdots \lambda_n$.*

En pratique, on se placera typiquement dans le cas $E = F = \mathbb{R}^2$ et on représentera f_i par $A_i \in \mathcal{M}_2(\mathbb{R})$ en utilisant les bases canoniques de $\mathbb{R}^2$ et $\mathcal{M}_2(\mathbb{R})$. L'intérêt principal de la notation tensorielle sera de nous permettre de représenter très simplement certaines matrices de grande dimension et de trouver facilement leurs éléments propres, là où une approche classique aurait en apparence une complexité rédhibitoire. Pour les calculs explicites, on pourra indexer la base $(e_{i_1} \otimes \cdots \otimes e_{i_n})_{1 \leqslant i_1, \dots, i_n \leqslant 2}$ en utilisant l'ordre lexicographique (où (e_1, e_2) est la base canonique de $\mathbb{R}^2$), ce qui correspond à définir $A_1 \otimes \cdots \otimes A_n$ comme le *produit de Kronecker* des matrices $A_1, \dots, A_n \in \mathcal{M}_2(\mathbb{R})$.

Chapitre 1

Un modèle physique pour décrire les gènes en interaction

Dans ce chapitre nous allons construire un modèle dynamique de réseau de régulation, c'est-à-dire décrivant l'évolution temporelle des niveaux d'expression d'un ensemble de gènes potentiellement en interaction. Comme évoqué dans l'introduction, l'idée est de partir du diagramme de la Figure 1 tout en ajoutant la possibilité d'une rétroaction des protéines sur la transcription. Le résultat sera un processus stochastique – prenant la forme d'un système de particules en interaction – et nous adopterons tantôt le point de vue des *trajectoires* (modèle à base d'agents), tantôt celui, dual, de la *distribution* (équation maîtresse).

1.1 Expression d'un gène isolé

Avant d'envisager la modélisation d'un système aussi complexe qu'un réseau de gènes, il est nécessaire de s'intéresser à la dynamique individuelle de chaque gène. On a besoin d'un modèle assez sophistiqué pour décrire l'évolution des quantités qui nous intéressent (ARNm et protéines), mais également assez simple pour être accessible mathématiquement. Les modélisateurs se penchent depuis longtemps sur la dynamique stochastique des gènes et il existe aujourd'hui de nombreux modèles (Boettiger, 2013). Leur complexité est très variable mais ils ont en commun un formalisme mathématique hérité de la chimie, où la dynamique temporelle des molécules est décrite par des processus markoviens de type naissance-mort. Puisque l'expression d'un gène se résume à un ensemble (très grand) de réactions chimiques, l'idée est de choisir quelles réactions on souhaite décrire précisément.

1.1.1 Modèle standard

Notre point de départ est le *modèle à deux états*,[1] qui est rapidement devenu standard car c'est le plus simple qui soit capable de décrire l'expression d'un gène de manière satisfaisante à l'échelle de la cellule (Raser et O'Shea, 2004 ; Becskei *et al.*, 2005 ; Raj *et al.*, 2006 ; Suter *et al.*, 2011). Il s'agit en fait de la version en temps continu d'une chaîne de Markov introduite par Ko (1991) sur la base d'expériences pionnières en cellules uniques (Ko *et al.*, 1990). Dans ce modèle, un gène est décrit par son *promoteur* qui peut être soit « actif » (ON) soit « inactif » (OFF), représentant par exemple l'état de fixation sur l'ADN d'un complexe

1. Il est aussi appelé *random telegraph* dans la littérature de biologie : nous n'utiliserons pas ce nom ici afin d'éviter la confusion avec des homonymes issus d'autres communautés scientifiques.

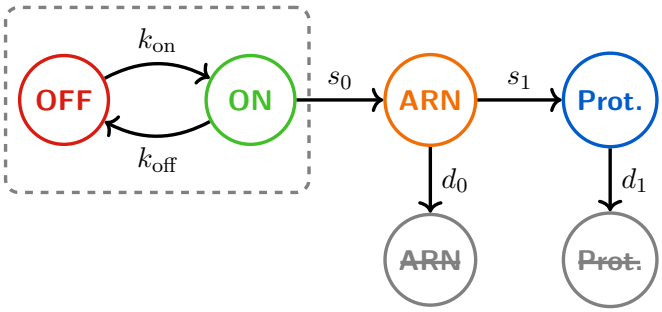


Figure 1.1 – Diagramme classique du modèle à deux états.

moléculaire d'initiation de la transcription. Le phénomène à l'origine de ces deux états peut être plus subtile (Larson, 2011 ; Chong *et al.*, 2014), mais le principe essentiel est que le gène ne produit de l'ARNm que lorsque le promoteur est actif. On ajoute l'étape de traduction, ainsi que la dégradation de l'ARNm et des protéines, comme des réactions de premier ordre. Plus précisément, le modèle correspond aux réactions élémentaires suivantes :

$$\begin{aligned} \mathsf{G} \xrightarrow{k_{\text{on}}} \mathsf{G}^*, \quad & \mathsf{G}^* \xrightarrow{k_{\text{off}}} \mathsf{G} \\ \mathsf{G}^* \xrightarrow{s_0} \mathsf{G}^* + \mathsf{M}, \quad & \mathsf{M} \xrightarrow{s_1} \mathsf{M} + \mathsf{P} \\ \mathsf{M} \xrightarrow{d_0} \varnothing, \quad & \mathsf{P} \xrightarrow{d_1} \varnothing \end{aligned} \tag{1.1}$$

où G, G^*, M et P désignent respectivement le promoteur inactif, le promoteur actif, l'ARNm et les protéines. Une représentation classique est le diagramme de la Figure 1.1. Celui-ci n'a pas vraiment de sens mathématiquement, mais il est souvent utilisé car il permet de décrire simplement les six réactions chimiques considérées ainsi que leurs taux (par molécule de chaque réactif) : chaque nœud correspond à une espèce et chaque flèche à une réaction. Dans toute la suite, on supposera que ces taux sont strictement positifs.

Il nous faut maintenant un modèle mathématique pour décrire le système de réactions chimiques (1.1). Une des caractéristiques fondamentales est que les espèces considérées sont présentes en très petite quantité par rapport au nombre d'Avogadro, sans compter que G et G^* sont par définition des espèces rares puisque leurs nombres de molécules $[\mathsf{G}]$ et $[\mathsf{G}^*]$ sont égaux à 0 ou 1 avec à tout instant $[\mathsf{G}] + [\mathsf{G}^*] = 1$. L'option typique consiste alors à supposer que toutes les réactions élémentaires suivent la *loi d'action des masses stochastique*, ce qui correspond, en notant respectivement $E_t = [\mathsf{G}^*]$, $M_t = [\mathsf{M}]$ et $P_t = [\mathsf{P}]$ à l'instant t, à dire que $(E_t, M_t, P_t)_{t \geqslant 0}$ est un processus markovien de sauts à valeurs dans $\mathcal{S} = \{0, 1\} \times \mathbb{N} \times \mathbb{N}$ dont le générateur L est donné, pour $m, p \in \mathbb{N}$ et $f = (f_0, f_1)^\top : \mathcal{S} \to \mathbb{R}$, par

$$\begin{aligned} Lf(m,p) = {} & Qf(m,p) \\ & + S[f(m+1,p) - f(m,p)] + s_1 m[f(m,p+1) - f(m,p)] \\ & + d_0 m[f(m-1,p) - f(m,p)] + d_1 p[f(m,p-1) - f(m,p)] \end{aligned} \tag{1.2}$$

avec

$$S = \begin{pmatrix} 0 & 0 \\ 0 & s_0 \end{pmatrix} \quad \text{et} \quad Q = \begin{pmatrix} -k_{\text{on}} & k_{\text{on}} \\ k_{\text{off}} & -k_{\text{off}} \end{pmatrix}.$$

Il existe alors une méthode de construction très classique des trajectoires, que l'on rappelle ici. Étant donné $(e, m, p) = (E_t, M_t, P_t) \in \mathcal{S}$ l'état du processus à l'instant t, on a six évènements

possibles récapitulés dans le tableau suivant, indexés arbitrairement par $i \in [\![1, 6]\!]$ et pouvant chacun survenir de manière indépendante avec un taux λ_i.

i	Taux λ_i	Interprétation	État d'arrivée
1	$k_{\text{on}}(1-e)$	Passage dans l'état actif	$(1-e, m, p)$
2	$k_{\text{off}}e$	Passage dans l'état inactif	$(1-e, m, p)$
3	$s_0 e$	Création d'un ARNm	$(e, m+1, p)$
4	$s_1 m$	Création d'une protéine	$(e, m, p+1)$
5	$d_0 m$	Dégradation d'un ARNm	$(e, m-1, p)$
6	$d_1 p$	Dégradation d'une protéine	$(e, m, p-1)$

On construit alors la trajectoire de la façon suivante : l'état du processus reste égal à (e, m, p) jusqu'à l'instant $t + T$ avec T aléatoire et distribué selon la loi exponentielle de paramètre $\lambda = \lambda_1 + \cdots + \lambda_6$, puis à cet instant le processus change d'état aléatoirement et arrive avec probabilité $p_i = \lambda_i/\lambda$ dans l'état correspondant à l'évènement i. On réitère ensuite ce procédé à partir de $t + T$, et ainsi de suite jusqu'à passer un instant final donné.

On peut vérifier que le processus $(E_t, M_t, P_t)_{t\geqslant 0}$ ainsi construit est bien un processus de Markov dont le semi-groupe a pour générateur l'opérateur L défini par (1.2), ce qui constitue un exercice classique (Liggett, 2010). On remarque bien sûr que chaque évènement de saut correspond précisément à l'une des réactions du système (1.1), et il se trouve que cette méthode de construction correspond au célèbre algorithme de simulation moléculaire introduit par Gillespie (1977). Noter qu'un physicien ou un biologiste dira directement qu'il simule (1.1) par l'algorithme de Gillespie.

Remarque 1.1. Dans le générateur (1.2), on a décrit M et P de manière traditionnelle mais on a « vectorisé » la notation pour G^*. Ce formalisme sera utilisé dans toute la suite et nous permettra de simplifier énormément les calculs.

1.1.2 Première simplification

Le processus de sauts défini par (1.2) est la façon la plus classique de modéliser un système chimique en distinguant chaque molécule.[2] En pratique, il n'est pas nécessaire de garder une description discrète pour M et P, qui sont des espèces abondantes et sans relation de conservation. Des expériences quantitatives (Schwanhäusser *et al.*, 2011) suggèrent que les paramètres de création et dégradation vérifient typiquement $s_0 \gg d_0$ et $s_1 \geqslant d_1$. Dans ce régime, l'échelle des niveaux d'ARNm et de protéines est assez grande pour qu'il semble tout à fait satisfaisant de négliger leur « bruit moléculaire » en les considérant comme des quantités continues qui suivent les équations différentielles associées à la loi d'action des masses classique (i.e. déterministe). On obtient le modèle hybride :

$$\begin{cases} E_t : 0 \xrightarrow{k_{\text{on}}} 1, \;\; 1 \xrightarrow{k_{\text{off}}} 0 \\ \dot{M}_t = s_0 E_t - d_0 M_t \\ \dot{P}_t = s_1 M_t - d_1 P_t \end{cases} \tag{1.3}$$

en utilisant une notation intuitive empruntée à Rudnicki (2015). En d'autres termes, l'état du promoteur E_t suit toujours le même processus markovien de sauts à deux états, mais l'ARNm

2. Pour une présentation générale du formalisme, on pourra par exemple consulter l'excellent et très concis Anderson et Kurtz (2015) où l'hypothèse prend le nom de *stochastic mass-action kinetics*.

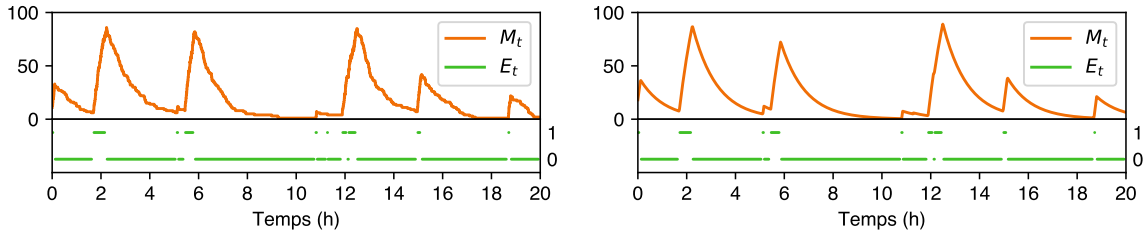


Figure 1.2 – Exemple de trajectoires du modèle discret (gauche) et du modèle PDMP (droite), où l'on ne représente ici que l'état du promoteur (E_t) et l'ARNm (M_t). Le fait de considérer la même trajectoire de E_t permet de se rendre compte du lien entre les deux processus. De manière intuitive, la production des protéines sera alors très similaire dans les deux cas. Les valeurs des paramètres choisies pour cet exemple sont $k_{\text{on}} = 0.6$, $k_{\text{off}} = 5$, $d_0 = 1$ et $s_0 = 200$ (en h^{-1}).

et les protéines suivent maintenant des équations différentielles ordinaires dont la première dépend de E_t (Figure 1.2). Ce modèle n'est autre qu'un processus de Markov déterministe par morceaux (PDMP) appartenant à la classe bien connue des *switching ODEs* (Benaïm *et al.*, 2012, 2015). Il y a deux façons de le justifier mathématiquement :

- On peut montrer que c'est bien la limite du modèle discret (1.2) dans le régime $s_0 \gg d_0$ et $s_1 \geqslant d_1$, ce qui a été fait par Crudu *et al.* (2012).
- On peut mettre en évidence un lien algébrique – donc exact – qui permet d'interpréter le PDMP comme la structure sous-jacente fondamentale du modèle discret. En particulier, une des conséquences probabilistes est que la trajectoire de l'ARNm provenant du PDMP apparaît comme l'espérance conditionnelle de son homologue dans le processus de sauts sachant la trajectoire du promoteur.

Le deuxième point sera détaillé dans le chapitre 3 où il aura un rôle essentiel. On renvoie également à la Figure 1.2 et au chapitre 6 pour une comparaison numérique. Pour l'heure, disons simplement que le modèle (1.3) est bien justifié dans notre cas : c'est ce modèle qui va nous servir de brique de base pour construire un réseau de régulation.

1.1.3 Le régime « bursty »

Lorsqu'à la fois $k_{\text{on}} \ll k_{\text{off}}$ et $d_0 \ll k_{\text{off}}$, l'ARNm est produit par *bursts*, c'est-à-dire pendant de courtes périodes de temps telles que son niveau reste toujours très en dessous de la saturation s_0/d_0 (cf. Figure 1.2). La quantité produite lors d'un burst s'approche alors au premier ordre comme étant $s_0 T$ où T est la durée du burst, qui suit par définition une loi exponentielle de paramètre k_{off}. De plus, les périodes actives du promoteur sont bien plus courtes que les périodes inactives de sorte que les premières peuvent être vues comme instantanées, ce qui justifie l'appellation *fréquence de burst* pour k_{on} qui est l'inverse de la durée moyenne de l'état inactif. Nous nous placerons dans ce régime dans toute la suite car c'est celui qui apparaît systématiquement dans les expériences (Raj *et al.*, 2006 ; Suter *et al.*, 2011 ; Viñuelas *et al.*, 2013 ; Albayrak *et al.*, 2016 ; Richard *et al.*, 2016).

Du point de vue mathématique (Crudu *et al.*, 2012), il est possible d'effectuer directement le passage à la limite $k_{\text{off}} \to +\infty$ à partir du modèle discret (1.2) en même temps que celui associé à $s_0 \gg d_0$, qui s'écrit alors $s_0 \to +\infty$ avec $\lambda = k_{\text{off}}/s_0$ fixé. L'état du promoteur n'a alors plus besoin d'être décrit puisque ses périodes d'activation sont infiniment courtes, et on obtient un nouveau processus $(M_t, P_t)_{t \geqslant 0}$ à valeurs dans $\mathcal{S} = \mathbb{R}_+ \times \mathbb{R}_+$ dont le générateur

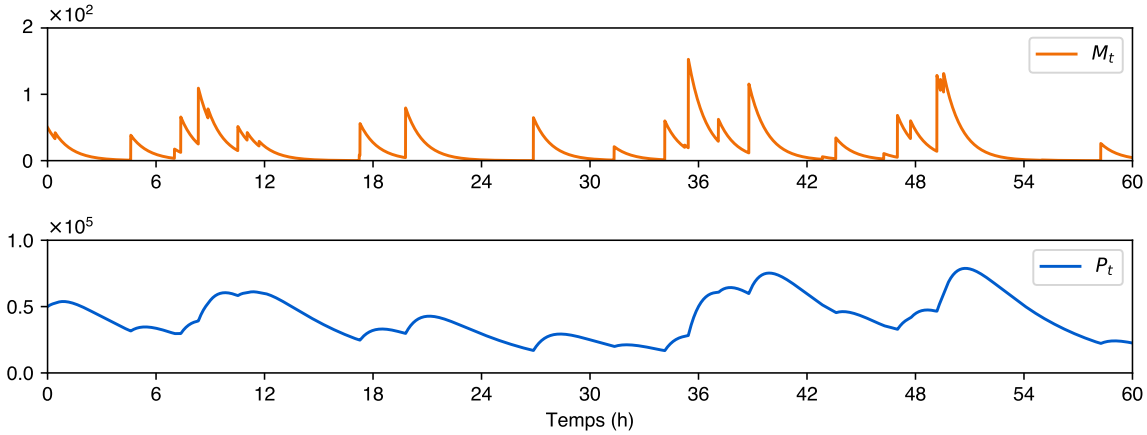


Figure 1.3 – Exemple de trajectoire du modèle PDMP bursty. Les valeurs des paramètres choisies sont $k_{\text{on}} = 0.6$, $k_{\text{off}} = 5$, $d_0 = 1$, $d_1 = 0.2$, $s_0 = 200$ et $s_1 = 400$ (en h^{-1}), le but étant d'obtenir des quantités réalistes d'ARNm et de protéines (Albayrak *et al.*, 2016).

est donné par

$$\begin{aligned} Lf(m,p) = k_{\text{on}} \int_0^\infty [f(m+y,p) - f(m,p)]\lambda e^{-\lambda y} dy \\ - d_0 m \partial_m f(m,p) + (s_1 m - d_1 p)\partial_p f(m,p) \end{aligned} \tag{1.4}$$

pour $f : \mathcal{S} \to \mathbb{R}$ de classe $\mathcal{C}^1$ et $(m,p) \in \mathcal{S}$. Il s'agit encore d'un PDMP, mais cette fois les sauts affectent directement le flot de l'équation différentielle et la trajectoire de M_t peut être choisie càdlàg mais pas continue. La construction d'une trajectoire se fait comme suit : étant donné $(m,p) = (M_t, P_t) \in \mathcal{S}$ l'état à l'instant t, le processus suit la dynamique déterministe donnée par

$$\begin{cases} \dot{M}_t = -d_0 M_t \\ \dot{P}_t = s_1 M_t - d_1 P_t \end{cases}$$

jusqu'à l'instant $t+T$ avec T distribué selon la loi exponentielle de paramètre k_{on}, puis à cet instant la première composante saute d'une hauteur Y (i.e. on pose $M_{t+T} = M_{t+T}^- + Y$ où M_{t+T}^- est la limite à gauche de $s \mapsto M_s$ en $t+T$) avec Y distribuée selon la loi exponentielle de paramètre λ. On répète alors ces deux étapes à partir de $t+T$, et ainsi de suite (Figure 1.3).

Remarquons finalement qu'il existe une variante assez populaire qui modélise seulement les protéines, de la même façon que l'ARNm ici (Friedman *et al.*, 2006 ; Mackey *et al.*, 2011 ; Pájaro *et al.*, 2017), avec une généralisation possible de la loi des hauteurs de saut. Nous n'utiliserons pas le modèle (1.4) dans la suite car il n'apporte pas réellement de simplification par rapport à (1.3) pour les problèmes qui nous intéressent, mais le régime bursty sera quand même intéressant à avoir en tête pour les applications. Par ailleurs, nous verrons que les résultats du chapitre 2 s'adaptent naturellement à ce régime limite.

1.1.4 Lois stationnaires

La loi stationnaire des protéines semble très difficile à calculer et à notre connaissance il n'existe toujours pas de forme explicite connue. En revanche, on peut calculer assez facilement la loi stationnaire de l'ARNm pour chacun des modèles (1.2), (1.3) et (1.4). Plus précisément,

en notant à chaque fois M une variable aléatoire distribuée selon cette loi et en posant

$$a = \frac{k_{\text{on}}}{d_0}, \quad b = \frac{k_{\text{off}}}{d_0}, \quad c = \frac{s_0}{d_0} \quad \text{et} \quad \lambda = \frac{k_{\text{off}}}{s_0},$$

on peut montrer que l'on a :

- $M \sim \text{Poisson}(cZ)$ où $Z \sim \text{Beta}(a, b)$ pour le processus de sauts (1.2) ;
- $M = cZ$ où $Z \sim \text{Beta}(a, b)$ pour le PDMP (1.3) ;
- $M \sim \text{Gamma}(a, \lambda)$ pour le PDMP bursty (1.4).

Les deux premiers points pourront s'interpréter comme un cas particulier du chapitre 3, et une preuve simple du troisième point est donnée par Malrieu (2015). On peut également vérifier facilement par la transformée de Laplace ou de Fourier que le troisième correspond bien à la limite en loi du deuxième lorsque $k_{\text{off}}, s_0 \to +\infty$ avec $\lambda = k_{\text{off}}/s_0$ fixé.

Notons que l'on retrouve la loi Gamma représentée sur la Figure 2 de l'introduction. Au passage, signalons que ce ne sont pas les deux états du promoteur qui pourraient induire la bimodalité observée après transformation des données : certes le modèle à deux états peut avoir une loi stationnaire bimodale (d'après les lois que l'on vient de donner, il faut et il suffit que k_{on} et k_{off} soient inférieurs à d_0), mais dans ce cas on a nécessairement une décroissance très rapide (de type Poisson) à partir du deuxième mode. Nous montrerons au chapitre 6 qu'une façon possible d'avoir la bimodalité observée est de considérer k_{on} non constant.

En fait, le modèle à deux états a été introduit pour remédier à l'incapacité de son ancêtre direct, le *modèle à un état*, de décrire les données de cellules uniques. On peut en effet voir ce dernier comme un cas particulier du modèle à deux états où le promoteur est en ON tout le temps (par exemple en fixant $k_{\text{off}} = 0$), ce qui aboutit au cas limite $Z = 1$ presque sûrement et ainsi $M \sim \text{Poisson}(c)$. Or les observations montrent systématiquement que la loi de Poisson est aberrante en comparaison de la loi Gamma (Halpern *et al.*, 2015 ; Albayrak *et al.*, 2016) ou de son alter ego discret, la loi binomiale négative, qui n'est autre que la loi mélange Poisson(Z) où Z suit une loi Gamma (Raj et van Oudenaarden, 2008).

1.2 Gènes en réseau et sous-modèle de chromatine

Pour le moment, on a seulement modélisé l'expression stochastique d'un gène isolé. En outre, si le sens physique de s_0, s_1, d_0 et d_1 est relativement clair, celui des taux k_{on} et k_{off} l'est moins car ces derniers résument en fait chacun un ensemble potentiellement très grand de réactions sous-jacentes autour de l'ADN. L'idée de la mise en réseau est de préciser ces réactions, ce qui nous permettra de définir explicitement des liens de cause à effet entre les gènes. L'approche que nous avons choisie pour décrire un ensemble de gènes en interaction est de ne plus considérer l'allumage et l'extinction comme des réactions élémentaires, mais plutôt comme des réactions *composées* qui font intervenir les protéines. Les notations $\mathsf{G} \to \mathsf{G}^*$ et $\mathsf{G}^* \to \mathsf{G}$ ne sont alors plus que des « équations-bilan » qu'il s'agit de décomposer pour obtenir la forme des taux de réaction k_{on} et k_{off}.

1.2.1 Modèle général de réseau

On considère à présent un ensemble de n gènes où $n \in \mathbb{N}^*$ et on note $\mathbb{R}_+^n = (\mathbb{R}_+)^n$. On va définir un processus $(E_t, M_t, P_t)_{t\geqslant 0}$ représentant le réseau, où cette fois

$$E_t = (E_{t,1}, \dots, E_{t,n}) \in \{0,1\}^n, \quad M_t = (M_{t,1}, \dots, M_{t,n}) \in \mathbb{R}_+^n \text{ et } P_t = (P_{t,1}, \dots, P_{t,n}) \in \mathbb{R}_+^n.$$

Pour $i \in [\![1, n]\!]$, les coordonnées du gène i sont donc $(E_{t,i}, M_{t,i}, P_{t,i})$ et on note $k_{\text{on},i}$, $k_{\text{off},i}$, $s_{0,i}$, $s_{1,i}$, $d_{0,i}$ et $d_{1,i}$ les taux des réactions du système (1.1) associés à ce gène. Notre hypothèse fondamentale est que ces taux sont constants à l'exception de $k_{\text{on},i}$ et $k_{\text{off},i}$, qui dépendent maintenant des protéines. En reprenant la dynamique donnée par (1.3), on a ainsi :

$$\forall i \in [\![1, n]\!], \quad \begin{cases} E_{t,i} : 0 \xrightarrow{k_{\text{on},i}(P_t)} 1, \quad 1 \xrightarrow{k_{\text{off},i}(P_t)} 0 \\ \dot{M}_{t,i} = s_{0,i} E_{t,i} - d_{0,i} M_{t,i} \\ \dot{P}_{t,i} = s_{1,i} M_{t,i} - d_{1,i} P_{t,i} \end{cases} \tag{1.5}$$

où $k_{\text{on},i}$ et $k_{\text{off},i}$ sont des fonctions de $\mathbb{R}_+^n$ dans $\mathbb{R}_+$ que l'on appellera *fonctions d'interaction*.

Remarque 1.2. On peut voir ce modèle comme un système de n particules en interaction probabiliste, où les particules sont décrites par l'espace d'états $\mathcal{S} = \{0,1\} \times \mathbb{R}_+ \times \mathbb{R}_+$. Ce point de vue sera utile dans la section 1.3 pour avoir une formulation simple du générateur.

Pour rester sur notre lancée mécaniste, la forme des fonctions d'interaction $k_{\text{on},i}$ et $k_{\text{off},i}$ doit découler d'un modèle biochimique sous-jacent des interactions promoteur-protéines. On pourra ainsi faire en sorte que le réseau de régulation soit représenté par un traditionnel graphe orienté, tout en sachant que chaque interaction est précisément définie.

Exemple 1.3. Si l'on suppose que les interactions proviennent des réactions élémentaires suivantes se produisant en parallèle :

$$\forall i \in [\![1, n]\!], \quad \begin{cases} \mathsf{G}_i \xrightarrow{k^0_{\text{on},i}} \mathsf{G}_i^* \\ \mathsf{G}_i^* \xrightarrow{k^0_{\text{off},i}} \mathsf{G}_i \end{cases} \quad \text{et} \quad \forall (i,j) \in [\![1, n]\!]^2, \quad \begin{cases} \mathsf{G}_i + 2\mathsf{P}_j \xrightarrow{\nu_{ij}} \mathsf{G}_i^* + 2\mathsf{P}_j \\ \mathsf{G}_i^* + 2\mathsf{P}_j \xrightarrow{\mu_{ij}} \mathsf{G}_i + 2\mathsf{P}_j \end{cases}$$

alors en appliquant la loi d'action des masses classique, on obtient la forme

$$k_{\text{on},i}(P) = k^0_{\text{on},i} + \sum_{j=1}^n \nu_{ij} {P_j}^2 \quad \text{and} \quad k_{\text{off},i}(P) = k^0_{\text{off},i} + \sum_{j=1}^n \mu_{ij} {P_j}^2$$

pour tout $P = (P_1, \dots, P_n) \in \mathbb{R}_+^n$. L'idée derrière les paramètres μ_{ij} et ν_{ij} est bien sûr de fournir une idée de réseau : seule une petite partie de ces paramètres devrait être non nulle et telle que $\mu_{ij}\nu_{ij} = 0$, de sorte que ceux-ci correspondent aux flèches d'un graphe orienté.

Typiquement, on choisira $k_{\text{on},i}$ et $k_{\text{off},i}$ continues sur $\mathbb{R}_+^n$ et minorées par une constante strictement positive, comme dans l'Exemple 1.3 lorsque $k^0_{\text{on},i}, k^0_{\text{off},i} > 0$. Cette hypothèse n'est pas très contraignante et permet, en utilisant des résultats spécifiquement adaptés à ce type de modèle, de s'assurer que le processus $(E_t, M_t, P_t)_{t\geqslant 0}$ est bien défini et que la loi de (E_t, M_t, P_t) converge vers une unique loi stationnaire (cf. section 1.3). Ainsi, nous appellerons *modèle de réseau* le processus $(E_t, M_t, P_t)_{t\geqslant 0}$ défini par (1.5) avec la donnée d'un choix particulier de fonctions d'interaction $k_{\text{on},i}$ et $k_{\text{off},i}$. La correspondance d'un tel réseau avec

un graphe orienté dépend fortement de la forme de ces fonctions : à cet égard, l'Exemple 1.3 est séduisant mais malheureusement pas réaliste : les simulations de ce modèle se révèlent complètement aberrantes par rapport aux données. En outre, il devient aujourd'hui évident que l'on ne peut pas se baser uniquement sur la loi d'action des masses classique, la régulation étant directement liée à la conformation spatiale de l'ADN dans la cellule et en particulier à l'état de la chromatine (Fourel *et al.*, 2004 ; Rao *et al.*, 2014 ; Haddad *et al.*, 2017). Il nous faut donc oublier la chimie classique et fabriquer une « nouvelle loi », par exemple à partir d'un cadre probabiliste bien choisi. Nous allons aborder cet aspect de manière très sommaire par un modèle abstrait, mais il semble que ce soit déjà un pas de plus dans le contexte de l'inférence de réseaux de régulation.

1.2.2 Un sous-modèle possible pour les interactions

Dans la suite de cette section, nous décrivons un modèle biochimique sous-jacent capable de fournir une forme explicite pour les fonctions d'interaction : pour un gène i fixé, le but est de définir les fonctions $k_{\mathrm{on},i}$ et $k_{\mathrm{off},i}$. Pour simplifier les notations, nous omettrons l'indice i lorsqu'il n'y a pas d'ambiguité. Nous ne considérons que des réactions de type *hit-and-run*, les protéines ne restant alors fixées que très peu de temps sur la chromatine, mais il n'y a pas de différence mathématique avec la situation où elles resteraient fixées plus longtemps[3] et il s'avère que ce modèle peut s'interpréter comme un cas particulier de *multisite binding* (Mirny, 2010 ; Mazza et Benaïm, 2014). De plus, cette partie est issue du document « Supplementary information » associé à notre article dans *BMC Systems Biology* (dont le texte principal sera présenté en détail au chapitre 6) et elle est donc rédigée en anglais.

Simple biochemical model

We consider a set of reversible transitions between some chromatin states (e.g. describing enhancer regions). Each chromatin state is then associated with a particular rate for the promoter activation reaction. For simplicity, we consider only two cases : a high rate k_1 (the chromatin will be said *permissive*) and a low rate $k_0 \ll k_1$ (the chromatin will be said *non-permissive*). Once active, the promoter can switch off at a rate that is supposed to be independent from chromatin states, representing for example the stability of a transcription initiation complex. Finally, we assume that such chromatin transitions correspond to fast interactions with ambiant proteins (binding, hit-and-run, etc.) so that the promoter-switching reactions always see chromatin in its quasi-steady state. Effective rates k_{on} and k_{off} can therefore be obtained by averaging over chromatin states : this way, k_{off} is still a constant and k_{on} is now defined by

$$k_{\mathrm{on}} = k_0 p_0 + k_1 p_1$$

where p_0 (resp. p_1) is the probability of chromatin being non-permissive (resp. permissive).

We now define an explicit model for chromatin dynamics and compute its stationary distribution to derive p_0 and p_1 as functions of protein levels $P_1, \dots, P_n$. We consider 2^n permissive configurations and 2^n non-permissive configurations as follows : for all $I \subset \mathcal{G}$ where $\mathcal{G} = [\![1, n]\!]$, species C_I (resp. C_I^*) stands for the chromatin being non-permissive (resp. permissive) and in state I. The underlying physics are the following : the chromatin has two "basal" configurations, $\mathsf{C}_\emptyset$ (non-permissive) and $\mathsf{C}_\emptyset^*$ (permissive), which describe dynamics

3. Leur concentration sera supposée constante à l'échelle de temps (plus rapide) du sous-modèle.

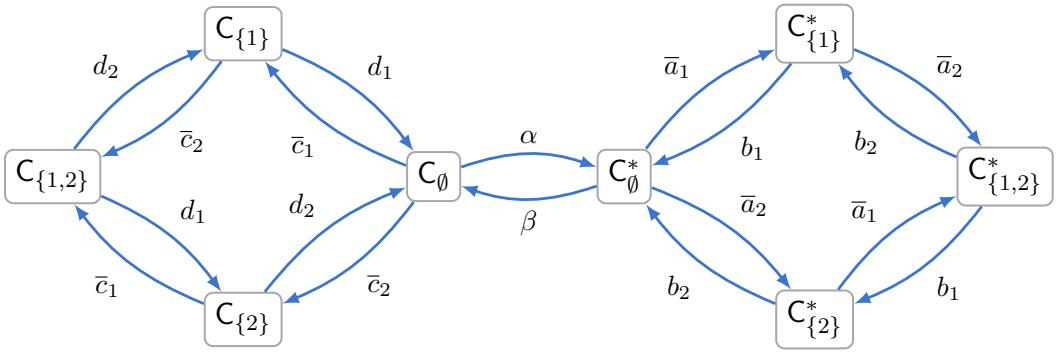


Figure 1.4 – Chromatin states and transitions in the case $n = 2$, with $\bar{a}_j = a_j[\mathsf{P}_j]$ and $\bar{c}_j = c_j[\mathsf{P}_j]$.

when no protein is present, according to the reactions

$$\mathsf{C}_\emptyset \xrightarrow{\alpha} \mathsf{C}^*_\emptyset, \quad \mathsf{C}^*_\emptyset \xrightarrow{\beta} \mathsf{C}_\emptyset. \tag{1.6}$$

Then, each protein P_j is able to modify the chromatin state through a "hit-and-run" reaction, which is kept in memory by encoding the index j in the list I, giving the state C_I or C^*_I. Eventually, this memory can be lost by "emptying" I step by step (going back to the basal configuration). That is, for all $I \subset \mathcal{G}$ and $j \in \mathcal{G} \setminus I$, we consider the reactions

$$\begin{aligned} \mathsf{C}^*_I + \mathsf{P}_j &\xrightarrow{a_j} \mathsf{C}^*_{I\cup j} + \mathsf{P}_j, \quad \mathsf{C}^*_{I\cup j} \xrightarrow{b_j} \mathsf{C}^*_I, \\ \mathsf{C}_I + \mathsf{P}_j &\xrightarrow{c_j} \mathsf{C}_{I\cup j} + \mathsf{P}_j, \quad \mathsf{C}_{I\cup j} \xrightarrow{d_j} \mathsf{C}_I. \end{aligned} \tag{1.7}$$

The system then evolves with $[\mathsf{C}_I], [\mathsf{C}^*_I] \in \{0,1\}$ and $\sum_I [\mathsf{C}_I] + [\mathsf{C}^*_I] = 1$, so that only one molecule is present at a time : its species therefore entirely describes the state of the system. Mathematically, we obtain a standard jump Markov process with 2^{n+1} states. For example, the case $n = 2$ leads to the scheme of Figure 1.4. The underlying idea is that, depending on a_j, b_j, c_j and d_j, proteins will tend to stabilize the chromatin either in a permissive configuration or in a non-permissive one, thus providing notions of *activation* and *inhibition*. The basal reactions with rates α and β sum up what we do not observe (i.e. what is likely to happen when none of the P_j is present).

Stationary distribution

Letting $\mathcal{S} = \{0,1\}^{n+1}$, each state can be coded by a vector $s = (s_0, s_1, \dots, s_n) \in \mathcal{S}$ where $s_0 = 1$ if the chromatin is permissive and 0 otherwise, and for $j \geqslant 1$, $s_j = 1$ if it has been modified by protein P_j and 0 otherwise. If all rates are positive, the system has a unique stationary distribution π which can be exactly computed from the master equation (see section 1.2.3 below). More precisely, the probability π_s of the chromatin being in state $s \in \mathcal{S}$ is given by

$$\pi_s = \begin{cases} Z^{-1}\alpha \prod_{j=1}^n (\lambda_j[\mathsf{P}_j]s_j + 1 - s_j) & \text{if } s_0 = 1 \\ Z^{-1}\beta \prod_{j=1}^n (\mu_j[\mathsf{P}_j]s_j + 1 - s_j) & \text{if } s_0 = 0 \end{cases} \tag{1.8}$$

where $\lambda_j = a_j/b_j$, $\mu_j = c_j/d_j$ and Z is a normalizing constant. Now going back to our initial intention of computing k_{on}, we are only interested in the probability for the chromatin to be

permissive,

$$p_1 = \sum_{s_1,\dots,s_n} \pi_{(1,s_1,\dots,s_n)} = Z^{-1}\alpha \sum_{s_1,\dots,s_n} \prod_{j=1}^{n} (\lambda_j[\mathsf{P}_j]s_j + 1 - s_j),$$

and the probability for the chromatin to be non-permissive,

$$p_0 = \sum_{s_1,\dots,s_n} \pi_{(0,s_1,\dots,s_n)} = Z^{-1}\beta \sum_{s_1,\dots,s_n} \prod_{j=1}^{n} (\mu_j[\mathsf{P}_j]s_j + 1 - s_j).$$

Observing that each product term only depends on one s_j, these formulas collapse to

$$p_1 = Z^{-1}\alpha \prod_{j=1}^{n} (\lambda_j[\mathsf{P}_j] + 1), \quad p_0 = Z^{-1}\beta \prod_{j=1}^{n} (\mu_j[\mathsf{P}_j] + 1)$$

and the distribution condition $p_0 + p_1 = 1$ gives $Z = \alpha \prod_{j=1}^{n}(\lambda_j[\mathsf{P}_j]+1) + \beta \prod_{j=1}^{n}(\mu_j[\mathsf{P}_j]+1)$. We finally get

$$k_{\text{on}} = \frac{k_0\beta \prod_{j=1}^{n}(\mu_j[\mathsf{P}_j] + 1) + k_1\alpha \prod_{j=1}^{n}(\lambda_j[\mathsf{P}_j] + 1)}{\beta \prod_{j=1}^{n}(\mu_j[\mathsf{P}_j] + 1) + \alpha \prod_{j=1}^{n}(\lambda_j[\mathsf{P}_j] + 1)}. \tag{1.9}$$

From this formula, it is straightforward to see that k_{on} will actually depend on a protein P_j only if $\lambda_j \neq \mu_j$, that is, when reactions involving P_j have unbalanced speeds and tend to favor either permissive configurations ($\lambda_j > \mu_j$) or non-permissive configurations ($\lambda_j < \mu_j$).

Higher order interactions

So far we only considered that the P_j were interacting as monomers. If they in fact interact after forming dimers or other complexes, and if such complex-forming reactions are even faster than chromatin dynamics, one can take this into account by replacing $[\mathsf{P}_j]$ in equation (1.9) with a function of $[\mathsf{P}_j]$ corresponding to the quasi-steady state concentration of the complex. This seems relevant to capture the overall dependence of k_{on} on the proteins, the main point being to use a continuous description for proteins, which are abundant, while keeping a discrete (stochastic) description for chromatin. We chose to replace $[\mathsf{P}_j]$ with $[\mathsf{P}_j]^{m_j}$ where $m_j \in \mathbb{R}_+$, which gives our model a general Hill-type form. Note that $m_j = 2$ (resp. $m_j = 3$) may represent a correct approximation for P_j interacting as a dimer (resp. a trimer) but in general m_j does not necessarily have to be an integer (Mazza et Benaïm, 2014).

The case of auto-activation

A this stage, it is possible to implement self-interaction for gene i by taking $\lambda_i \neq \mu_i$ in (1.9) but this leads to obvious identifiability issues : in stationary state, one cannot really distinguish between auto-activation, auto-inhibition and basal level. To cope with these, we restrict ourselves to auto-activation by setting $c_i = d_i = 0$ and keeping only the relevant chromatin states (C_I^* for all I, and C_I for I such that $i \notin I$). The system still has a unique stationary distribution and the formula for k_{on} corresponds to the case $\mu_i = 0$ in (1.9). Then, starting from the fact that auto-activation is only relevant when the basal level is small enough (for a bistable behaviour to be possible), we take the limit $\alpha \ll 1$ while keeping $\alpha\lambda_i$

fixed : the formula becomes

$$k_{\text{on}} = \frac{k_0\beta \prod_{j\neq i}(\mu_j[\mathsf{P}_j]^{m_j}+1) + k_1\alpha\lambda_i[\mathsf{P}_i]^{m_i}\prod_{j\neq i}(\lambda_j[\mathsf{P}_j]^{m_j}+1)}{\beta \prod_{j\neq i}(\mu_j[\mathsf{P}_j]^{m_j}+1) + \alpha\lambda_i[\mathsf{P}_i]^{m_i}\prod_{j\neq i}(\lambda_j[\mathsf{P}_j]^{m_j}+1)} \tag{1.10}$$

where $m_i > 0$ if gene i activates itself and $m_i = 0$ otherwise.

Parameterization for inference

Parameters of equation (1.10) are still clearly not identifiable : in order to get a more minimal form, we introduce the following parameterization : $s_j = {\mu_j}^{-1/m_j}$, $\theta_j = \log(\lambda_j/\mu_j)$ for all $j \neq i$, and $s_i = (\beta/\alpha)^{1/m_i}$, $\theta_i = \log(\lambda_i)$. After simplifying (1.10), we obtain

$$k_{\text{on}} = \frac{k_0 + k_1\Phi}{1+\Phi}$$

where

$$\Phi = e^{\theta_i}([\mathsf{P}_i]/s_i)^{m_i}\prod_{j\neq i}\frac{1+e^{\theta_j}([\mathsf{P}_j]/s_j)^{m_j}}{1+([\mathsf{P}_j]/s_j)^{m_j}}.$$

The new parameters have an intuitive meaning : s_j can be seen as a threshold for the influence by protein j, and θ_j characterizes this influence via its sign and absolute value ($\theta_j = 0$ implying that k_{on} does not depend on protein j), with the exception that s_i and θ_i aggregate a basal behaviour and an auto-activation strength.

Finally, we recall the notation $P_j = [\mathsf{P}_j]$ and reintroduce the index i of the gene of interest and add it to each parameter. Hence, for every gene i, the function $k_{\text{on},i}$ is defined by :

$$k_{\text{on},i}(P) = \frac{k_{0,i} + k_{1,i}\Phi_i(P)}{1+\Phi_i(P)} \tag{1.11}$$

with

$$\Phi_i(P) = e^{\theta_{ii}}(P_i/s_{ii})^{m_{ii}}\prod_{j\neq i}\frac{1+e^{\theta_{ij}}(P_j/s_{ij})^{m_{ij}}}{1+(P_j/s_{ij})^{m_{ij}}}. \tag{1.12}$$

In our statistical framework, we shall assume that parameters $k_{0,i}$, $k_{1,i}$, m_{ij} and s_{ij} are known and we focus on inferring the matrix $\theta = (\theta_{ij}) \in \mathbb{R}^{n\times n}$, which is similar to the interaction matrix in usual gene network inference methods.

1.2.3 Précisions

Donnons quelques précisions sur la façon dont on obtient la loi stationnaire (1.8) de ce modèle de chromatine. Comme nous l'avions évoqué à la fin de l'introduction, on va exploiter le formalisme des produits tensoriels. Rappelons que chaque état est codé par un vecteur $s = (s_0, s_1, \ldots, s_n) \in \mathcal{S} = \{0,1\}^{n+1}$ où :

- $s_0 = 1$ si la chromatine est permissive et 0 sinon ;
- pour $i \in [\![1, n]\!]$, $s_i = 1$ si la chromatine a été modifiée par la protéine P_i et 0 sinon.

On a un espace d'états $\mathcal{S}$ sans structure particulière, alors que la dynamique donnée par le système de réactions (1.6)-(1.7) a l'air d'être assez simple dans le sens où l'on ne peut pas passer d'un état de $\mathcal{S}$ à n'importe quel autre (cf. Figure 1.4). En fait, l'étape cruciale est de trouver une convention *vectorielle* pour représenter à la fois $\mathcal{S}$ et la dynamique du processus.

L'ensemble $\mathcal{S}$ est de taille 2^{n+1} : il est naturel de représenter ses éléments par les vecteurs de la base canonique de $(\mathbb{R}^2)^{\otimes(n+1)}$, en notant cette base $(e_s)_{s\in\mathcal{S}}$ et en utilisant la convention :

$$e_s = \bigotimes_{i=0}^{n} \begin{pmatrix} \mathbb{1}_{\{0\}}(s_i) \\ \mathbb{1}_{\{1\}}(s_i) \end{pmatrix} = \bigotimes_{i=0}^{n} \begin{pmatrix} 1-s_i \\ s_i \end{pmatrix}, \quad \forall s \in \mathcal{S}.$$

Maintenant que l'on a une structure tensorielle pour $\mathcal{S}$, il s'agit de représenter le générateur de ce processus sous forme d'un élément de $(\mathcal{M}_2(\mathbb{R}))^{\otimes(n+1)}$. On considère en fait la transposée du générateur (parfois appelée *hamiltonien*) notée H, de façon à ce que l'équation de Kolmogorov progressive – ou *master equation* – décrivant l'évolution de la distribution π du processus représentée par $\pi = (\pi_s)_{s\in\mathcal{S}} \in (\mathbb{R}^2)^{\otimes(n+1)}$, s'écrive

$$\frac{d\pi}{dt} = H\pi.$$

Notre problème se réduira alors à trouver π tel que $H\pi = 0$ (on vérifie que H est irréductible donc on a bien une unique loi stationnaire). Il se trouve que l'on peut vérifier assez facilement que

$$H = H_\emptyset + \sum_{i=1}^{n} H_{i,0} + H_{i,1}$$

où

$$H_\emptyset = \begin{pmatrix} -\alpha & \beta \\ \alpha & -\beta \end{pmatrix} \otimes \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \otimes \cdots \otimes \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$$

$$H_{i,0} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \otimes I_2 \otimes \cdots \otimes I_2 \otimes \underbrace{\begin{pmatrix} -c_i[\mathsf{P}_i] & d_i \\ c_i[\mathsf{P}_i] & -d_i \end{pmatrix}}_{\text{rang } i} \otimes I_2 \otimes \cdots \otimes I_2$$

$$H_{i,1} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \otimes I_2 \otimes \cdots \otimes I_2 \otimes \underbrace{\begin{pmatrix} -a_i[\mathsf{P}_i] & b_i \\ a_i[\mathsf{P}_i] & -b_i \end{pmatrix}}_{\text{rang } i} \otimes I_2 \otimes \cdots \otimes I_2$$

avec I_2 la matrice identité de $\mathcal{M}_2(\mathbb{R})$. Intuitivement, les matrices dans $H_\emptyset$ situées après la première signifient que les transitions ne peuvent se faire que lorsque $s_i = 0$ pour tout $i \in [\![1, n]\!]$. De même, les transitions associées à $H_{i,0}$ (resp. $H_{i,1}$) se font lorsque $s_0 = 0$ (resp. $s_0 = 1$). Il ne reste plus qu'à trouver un vecteur propre pour la valeur propre 0. En regardant bien cette forme (il s'agit de combiner les vecteurs propres de $H_{i,0}$ et $H_{i,1}$), on obtient

$$\pi \propto \alpha \bigotimes_{i=0}^{n} u_i + \beta \bigotimes_{i=0}^{n} v_i$$

où $u_0 = (0,1)^\top$ et $v_0 = (1,0)^\top$ et pour tout $i \in [\![1, n]\!]$, $u_i = (1, \lambda_i[\mathsf{P}_i])^\top$ et $v_i = (1, \mu_i[\mathsf{P}_i])^\top$ avec $\lambda_i = a_i/b_i$ et $\mu_i = c_i/d_i$. En développant, on a finalement :

$$\pi_s = \begin{cases} Z^{-1}\alpha \prod_{i=1}^{n} (\lambda_j[\mathsf{P}_j]s_i + 1 - s_i) & \text{si } s_0 = 1 \\ Z^{-1}\beta \prod_{i=1}^{n} (\mu_i[\mathsf{P}_j]s_i + 1 - s_i) & \text{si } s_0 = 0 \end{cases}$$

qui correspond bien à (1.8).

Remarque 1.4. On peut vérifier que le processus de sauts défini par l'hamiltonien H est en fait réversible par rapport à π. En d'autres termes, le régime stationnaire correspond à un système physique en *équilibre thermodynamique*, c'est-à-dire sans flux d'énergie effectif lors des transitions entre les états. En pratique, on peut donc déterminer π directement à partir de l'équation *detailed balance* ($\pi_s Q_{s,s'} = \pi_{s'} Q_{s',s}$ pour tout $(s, s') \in \mathcal{S}^2$ où $Q = H^\top$), méthode très prisée en mécanique statistique (Mazza et Benaïm, 2014). Notre méthode vectorielle paraît a fortiori un peu inutilement technique, mais elle aurait l'avantage de pouvoir s'adapter à des variantes du modèle pour lesquelles le processus de sauts serait irréversible – on parle alors de système physique *hors équilibre* – sans doute plus réalistes (Coulon *et al.*, 2013).

1.3 Modèle final et exemples

Nous avons donc un modèle de réseau défini par $k_{\text{off},i}$ constante et $k_{\text{on},i}$ donnée par les équations (1.11) et (1.12). Ce modèle est plus réaliste que l'Exemple 1.3 puisque $k_{\text{on},i}$ (c'est-à-dire la « fréquence » des bursts) est bornée, et en pratique il permet bien de reproduire nos données d'expression. De plus, il est paramétré par une matrice carrée $\theta = (\theta_{ij})_{1 \leqslant i,j \leqslant n}$ dont on peut facilement vérifier que les coefficients correspondent au comportement logique attendu en représentant cette matrice par un graphe orienté pondéré :

- lorsque $\theta_{ij} = 0$, $k_{\text{on},i}$ ne dépend pas de P_j (pas d'interaction $j \to i$) ;
- lorsque $\theta_{ij} > 0$, $k_{\text{on},i}$ est une fonction croissante de P_j (activation $j \to i$) ;
- lorsque $\theta_{ij} < 0$, $k_{\text{on},i}$ est une fonction décroissante de P_j (inhibition $j \to i$).

C'est ce modèle qui sera utilisé dans les simulations, mais on peut garder à l'esprit le fait que $k_{\text{on},i}$ et $k_{\text{off},i}$ peuvent être des fonctions quelconques du moment qu'elles vérifient les hypothèses de la Proposition 1.6 ci-dessous. Le but étant de manipuler mathématiquement le PDMP, nous allons d'abord faire un changement d'échelle pour simplifier les notations, puis nous donnerons le générateur du processus ainsi obtenu.

1.3.1 Adimensionnement du modèle

On va s'affranchir des constantes qui n'ont pas un rôle structurel dans le modèle (1.5), ce qui va simplement correspondre à adimensionner les niveaux d'ARNm et de protéines. Dans cette section nous détaillons cet adimensionnement, que l'on appellera aussi « normalisation ». Pour commencer, considérons les ensembles

$$\mathcal{X}_E = \{0,1\}^n, \quad \mathcal{X}_M = \prod_{i=1}^n \left]0, \frac{s_{0,i}}{d_{0,i}}\right[\quad \text{et} \quad \mathcal{X}_P = \prod_{i=1}^n \left]0, \frac{s_{0,i}s_{1,i}}{d_{0,i}d_{1,i}}\right[.$$

On vérifie facilement qu'étant donnée une condition initiale $(E^0, M^0, P^0) \in \mathcal{X}_E \times \mathcal{X}_M \times \mathcal{X}_P$, c'est-à-dire $(E_1^0, \dots, E_n^0) \in \mathcal{X}_E$, $(M_1^0, \dots, M_n^0) \in \mathcal{X}_M$ et $(P_1^0, \dots, P_n^0) \in \mathcal{X}_P$, le processus $(E_t, M_t, P_t)_{t \geqslant 0}$ reste dans $\mathcal{X}_E \times \mathcal{X}_M \times \mathcal{X}_P$ pour tout $t \geqslant 0$. Les bornes supérieures des intervalles définissant $\mathcal{X}_M$ et $\mathcal{X}_P$ sont précisément les niveaux de saturation, respectivement de l'ARNm et des protéines. On introduit alors les variables adimensionnées

$$Y_{t,i} = \frac{d_{0,i}}{s_{0,i}} M_{t,i} \in]0,1[\quad \text{et} \quad X_{t,i} = \frac{d_{0,i}d_{1,i}}{s_{0,i}s_{1,i}} P_{t,i} \in]0,1[\quad \text{pour } i \in [\![1,n]\!]. \tag{1.13}$$

Comme $s_{0,i}$, $s_{1,i}$, $d_{0,i}$ et $d_{1,i}$ sont des constantes, on a d'après (1.5) :

$$\dot{Y}_{t,i} = \frac{d_{0,i}}{s_{0,i}}\dot{M}_{t,i} = d_{0,i}\left(E_{t,i} - \frac{d_{0,i}}{s_{0,i}}M_{t,i}\right) = d_{0,i}\left(E_{t,i} - Y_{t,i}\right)$$

et

$$\dot{X}_{t,i} = \frac{d_{0,i}d_{1,i}}{s_{0,i}s_{1,i}}\dot{P}_{t,i} = d_{1,i}\left(\frac{d_{0,i}}{s_{0,i}}M_{t,i} - \frac{d_{0,i}d_{1,i}}{s_{0,i}s_{1,i}}P_{t,i}\right) = d_{1,i}\left(Y_{t,i} - X_{t,i}\right).$$

On obtient finalement le modèle normalisé :

$$\forall i \in [\![1,n]\!], \quad \begin{cases} E_{t,i} : 0 \xrightarrow{a_i(X_t)} 1, \;\; 1 \xrightarrow{b_i(X_t)} 0 \\ \dot{Y}_{t,i} = d_{0,i}(E_{t,i} - Y_{t,i}) \\ \dot{X}_{t,i} = d_{1,i}(Y_{t,i} - X_{t,i}) \end{cases} \tag{1.14}$$

où a_i et b_i sont les fonctions d'interaction normalisées définies par

$$a_i(x) = k_{\mathrm{on},i}\left(\sigma_1 x_1, \ldots, \sigma_n x_n\right) \quad \text{et} \quad b_i(x) = k_{\mathrm{off},i}\left(\sigma_1 x_1, \ldots, \sigma_n x_n\right) \tag{1.15}$$

pour $x \in]0,1[^n$, où l'on a noté $\sigma_i = (s_{0,i}s_{1,i})/(d_{0,i}d_{1,i})$ le facteur d'échelle de la protéine i. Enfin, on vérifie qu'étant donnée une trajectoire $(E_t, Y_t, X_t)_{t\geqslant 0}$ du processus normalisé (1.14), la trajectoire $(E_t, M_t, P_t)_{t\geqslant 0}$ du processus original s'obtient simplement en inversant (1.13) : comme on a pris en compte l'adimensionnement dans a_i et b_i, les modèles (1.5) et (1.14) sont bien équivalents.

Remarque 1.5. Outre les fonctions a_i et b_i, seuls les paramètres $d_{0,i}$ et $d_{1,i}$ sont restés dans le modèle. Cela confirme que $s_{0,i}$ et $s_{1,i}$ ne sont que des facteurs d'échelle, mais que les taux de degradation – qui correspondent aux demi-vies des molécules – ont un rôle crucial dans la dynamique du réseau. Si par exemple un gène i est beaucoup plus « lent » que les autres ($d_{0,i}$ ou $d_{1,i}$ très faible), celui-ci va ralentir la dynamique partout où il a des interactions sortantes (les gènes j pour lesquels $\theta_{ji} \neq 0$). Notamment, le rapport $d_{0,i}/d_{1,i}$ aura un intérêt particulier qui sera étudié au chapitre suivant.

1.3.2 Formulation mathématique

Le processus normalisé $(E_t, Y_t, X_t)_{t\geqslant 0}$ sera notre point départ pour l'approche statistique. Nous allons donner son générateur sous une forme simple à partir de (1.14), puis nous en déduirons l'équation de Kolmogorov progressive associée – où équation maîtresse – décrivant l'évolution de la distribution, qui sera notre principal objet d'étude.

Soient $\mathcal{E} = \{0,1\}^n$, $\Omega =]0,1[^n$ et $\mathcal{S} = \mathcal{E} \times \Omega \times \Omega$. On notera $E_t = e = (e_1, \ldots, e_n) \in \mathcal{E}$, $Y_t = y = (y_1, \ldots, y_n) \in \Omega$ et $X_t = x = (x_1, \ldots, x_n) \in \Omega$ les valeurs prises par le processus. Comme on l'a vu avec le modèle de chromatine, le cadre tensoriel s'avère très pratique. On considère cette fois la base canonique de $(\mathbb{R}^2)^{\otimes n}$ notée $(r_e)_{e\in\mathcal{E}}$ avec la convention :

$$r_e = \bigotimes_{i=1}^{n} \begin{pmatrix} 1 - e_i \\ e_i \end{pmatrix}, \quad \forall e \in \mathcal{E}.$$

Pour $(y,x) \in \Omega \times \Omega$ et $i \in [\![1,n]\!]$, on définit $F_i(y_i), Q_i(x) \in \mathcal{M}_2(\mathbb{R})^{\otimes n}$ par

$$F_i(y_i) = I_2 \otimes \cdots \otimes \underbrace{F^{(i)}(y_i)}_{\text{rang } i} \otimes \cdots \otimes I_2, \qquad Q_i(x) = I_2 \otimes \cdots \otimes \underbrace{Q^{(i)}(x)}_{\text{rang } i} \otimes \cdots \otimes I_2$$

avec

$$F^{(i)}(y_i) = \begin{pmatrix} -d_{0,i}y_i & 0 \\ 0 & d_{0,i}(1-y_i) \end{pmatrix} \quad \text{et} \quad Q^{(i)}(x) = \begin{pmatrix} -a_i(x) & a_i(x) \\ b_i(x) & -b_i(x) \end{pmatrix},$$

et $G_i(y_i,x_i) = d_{1,i}(y_i - x_i)$. Le générateur est alors donné par

$$Lf = \sum_{i=1}^{n} F_i \partial_{y_i} f + G_i \partial_{x_i} f + Q_i f \tag{1.16}$$

pour $f = (f_e)_{e\in\mathcal{E}} : \mathcal{S} \to \mathbb{R}$ de classe $\mathcal{C}^1$, représentée semi-vectoriellement comme dans la définition (1.2) du modèle discret. Remarquons que cette expression sous forme de somme correspond bien au point de vue des systèmes de particules en interaction (Liggett, 2005). Dans le cas particulier où a_i et b_i sont constantes, les opérateurs $F_i\partial_{y_i} + G_i\partial_{x_i} + Q_i$ commutent entre eux, ce qui est facile à voir dans l'écriture tensorielle et traduit la situation où les gènes se comportent indépendamment les uns des autres. Enfin, on pose

$$K_i = Q_i^\top = I_2 \otimes \cdots \otimes \underbrace{K^{(i)}}_{\text{rang } i} \otimes \cdots \otimes I_2 \quad \text{où} \quad K^{(i)}(x) = Q^{(i)}(x)^\top = \begin{pmatrix} -a_i(x) & b_i(x) \\ a_i(x) & -b_i(x) \end{pmatrix}$$

et on considère $u : (t,y,x) \mapsto u(t,y,x) = (u_e(t,y,x))_{e\in\mathcal{E}}$ la distribution jointe de (E_t, Y_t, X_t), typiquement sous forme d'une densité. L'équation maîtresse vérifiée par u s'écrit alors

$$\partial_t u + \sum_{i=1}^{n} [\partial_{y_i}(F_i u) + \partial_{x_i}(G_i u)] = \sum_{i=1}^{n} K_i u \tag{1.17}$$

à interpréter a priori au sens des distributions. Remarquons que c'est un système d'équations de transport couplées, de dimension $|\mathcal{E}| = 2^n$. Pour tout $i \in [\![1,n]\!]$, le membre de gauche $u \mapsto \partial_{y_i}(F_i u) + \partial_{x_i}(G_i u)$ est un opérateur de transport pur (et il est diagonal du point de vue « vectoriel » i.e. de $\mathcal{E}$), tandis que le terme de droite peut se voir comme le Laplacien du graphe associé aux transitions entre les états des promoteurs.

Signalons enfin le résultat d'ergodicité suivant, qui est une application directe de (Benaïm *et al.*, 2015, Théorème 4.6). Nous esquissons la preuve dans le cas adimensionné, mais d'après ce qui précède ce résultat s'applique de la même façon au processus d'origine $(E_t, M_t, P_t)_{t\geqslant 0}$ et l'hypothèse peut se formuler indifféremment sur $k_{\text{on},i}$ et $k_{\text{off},i}$ ou sur a_i et b_i.

Proposition 1.6. *On suppose que pour tout $i \in [\![1,n]\!]$, les fonctions $k_{\text{on},i}$ et $k_{\text{off},i}$ sont continues sur $\mathbb{R}_+^n$ et minorées par des constantes strictement positives. Alors le processus $(E_t, Y_t, X_t)_{t\geqslant 0}$ est bien un PDMP ergodique, et $\mathcal{L}(E_t, Y_t, X_t)$ converge à vitesse exponentielle pour la distance en variation totale vers une unique loi stationnaire.*

Démonstration. Remarquons d'abord que l'on est bien dans le cadre de départ de Benaïm *et al.* (2015) puisque les taux de saut a_i et b_i définis par (1.15) sont des fonctions continues sur $[0,1]^n$ et minorées par des constantes strictement positives (les mêmes que pour $k_{\text{on},i}$ et

$k_{\text{off},i}$), ce qui implique que la matrice de transition $Q(x) = \sum_{i=1}^{n} Q_i(x)$ est irréductible pour tout $x \in [0,1]^n$. Ceci permet déjà de s'assurer que le processus $(E_t, Y_t, X_t)_{t\geqslant 0}$ est bien un PDMP. Dans ce qui suit, on note $\Gamma = [0,1]^n \times [0,1]^n$: il est assez facile de voir que c'est l'*ensemble accessible* de la composante continue $(X_t, Y_t)_{t\geqslant 0}$ correspondant au théorème 4.6 de Benaïm *et al.* (2015) dans notre cas. Pour appliquer ce théorème, il suffit de montrer qu'il existe un point $(x, y) = (x_1, \dots, x_n, y_1, \dots, y_n) \in \Gamma$ où la *strong bracket condition* est vérifiée par la famille de champs de vecteurs

$$\Big\{(x,y) \mapsto (d_{1,1}(y_1 - x_1), \dots, d_{1,n}(y_n - x_n), d_{0,1}(e_1 - y_1), \dots, d_{0,n}(e_n - y_n)),\ e \in \{0,1\}^n\Big\}.$$

Sans détailler les calculs qui sont assez lourds, on constate en fait rapidement que cette condition est vérifiée en tout point de Γ, en consultant par exemple Rudnicki (2015) pour la définition du crochet de Lie de deux champs de vecteurs (noter qu'une seule itération suffit). On en déduit ainsi l'ergodicité de $(E_t, Y_t, X_t)_{t\geqslant 0}$ et la convergence à vitesse exponentielle. □

1.3.3 Exemples

Pour donner un premier exemple à la fois simple et non trivial, on se place dans le cas $n = 3$ avec une voie de signalisation positive, c'est-à-dire d'activations successives :

où le sommet 0 correspond à un stimulus qui active le gène 1 à $t = 0$. À cet instant initial, tous les niveaux sont fixés à des valeurs très faibles, pour représenter une voie préalablement « fermée » qui s'ouvre à $t = 0$. On remarquera notamment que la variance peut être très forte d'une cellule à l'autre, les gènes suivant bien la logique attendue dans l'ordre d'expression, mais le temps de chaque déclenchement étant très variable (Figure 1.5 et Figure 1.6). À titre de comparaison, la Figure 1.7 montre une trajectoire de la version *bursty* basée sur (1.4) dans les mêmes conditions et pour les mêmes fonctions d'interaction. La Figure 1.8 montre un autre exemple de voie de signalisation, cette fois avec branchement.

Il convient de noter que l'on utilise à chaque fois une méthode de simulation exacte basée sur le principe du *thinning* : on saute en permanence au taux maximal, quitte à ce que certains sauts ne fassent pas changer d'état (*phantom jumps*). Ces sauts triviaux constituent une perte de temps pour la simulation, mais en échange la méthode est très simple à implémenter et ne nécessite aucune intégration numérique (Benaïm *et al.*, 2015), ce qui se révèle extrêmement pratique dans notre contexte où le nombre de variables peut être très grand. De ce point de vue, le modèle PDMP bursty est avantageux (dans le régime approprié) car il permet de réduire le taux de saut de façon substantielle.

On pourra aussi remarquer que l'on s'est placé dans un régime assez particulier, celui où les protéines sont lentes par rapport aux promoteurs et aux ARNm. En particulier, on a fixé $d_0/d_1 = 5$ pour tous les gènes, ce qui correspond à la valeur moyenne obtenue sur un grand nombre de gènes par Schwanhäusser *et al.* (2011). Les protéines jouent alors le rôle de filtre passe-bas en lissant les variations rapides de l'ARNm provenant de celles du promoteur, de sorte que seul l'état moyen de ce dernier sur une certaine fenêtre de temps a une influence sur l'évolution du niveau de protéines. En particulier, dans le régime bursty, c'est plutôt la fréquence d'activation k_{on} qui porte l'information : c'est précisément ce régime que nous considérons au chapitre suivant.

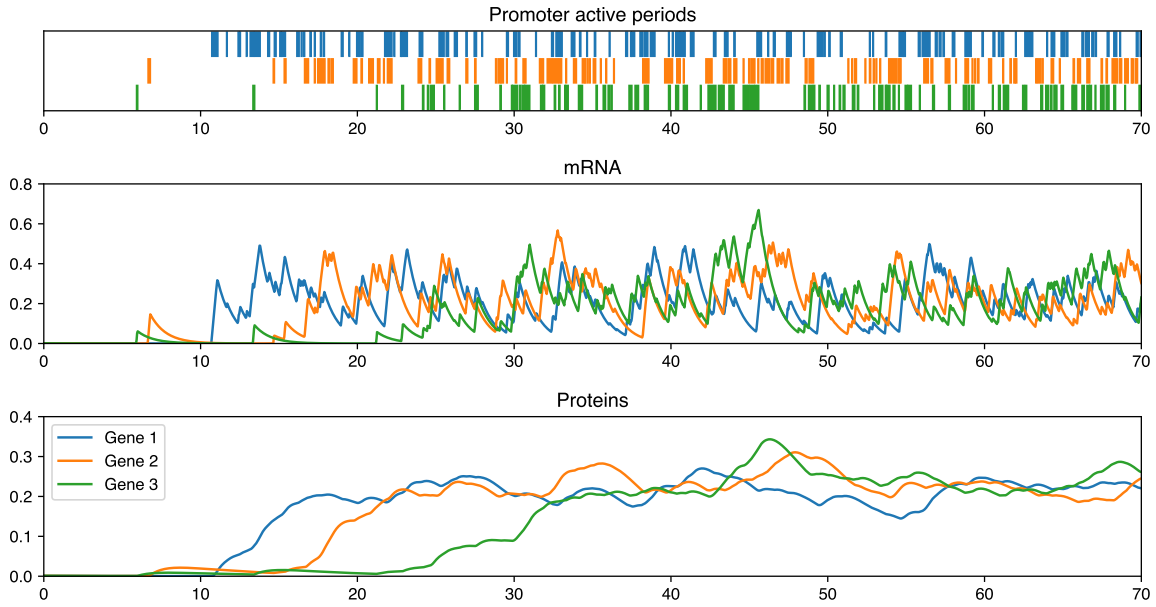


Figure 1.5 – Trajectoire d'une cellule pour un modèle de réseau défini par (1.14) et correspondant à la voie de signalisation $1 \to 2 \to 3$. Paramètres de base : $d_{0,i} = 1$, $d_{1,i} = 0.2$, $k_{0,i} = 0.1$, $k_{1,i} = 3.1$, $k_{\text{off},i} = 10$, $m_{ij} = 3$ et $s_{ij} = 0.12$ pour tout $i, j \in [\![1, 3]\!]$. Interactions : $\theta_{2,2} = \theta_{3,3} = 0.18$, $\theta_{1,1} = 5.18$ pour représenter le stimulus et $\theta_{2,1} = \theta_{3,2} = 5$, tous les autres θ_{ij} valant 0.

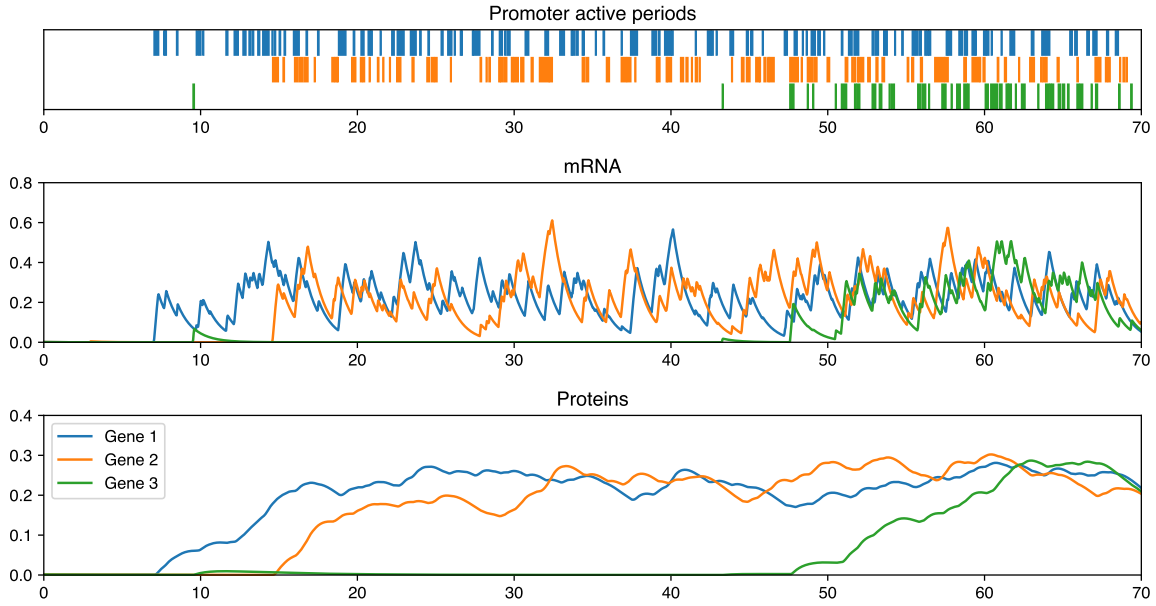


Figure 1.6 – Une autre trajectoire issue du même modèle que pour la Figure 1.5 et avec les mêmes conditions initiales. On remarque de manière générale une grande variance entre les trajectoires.

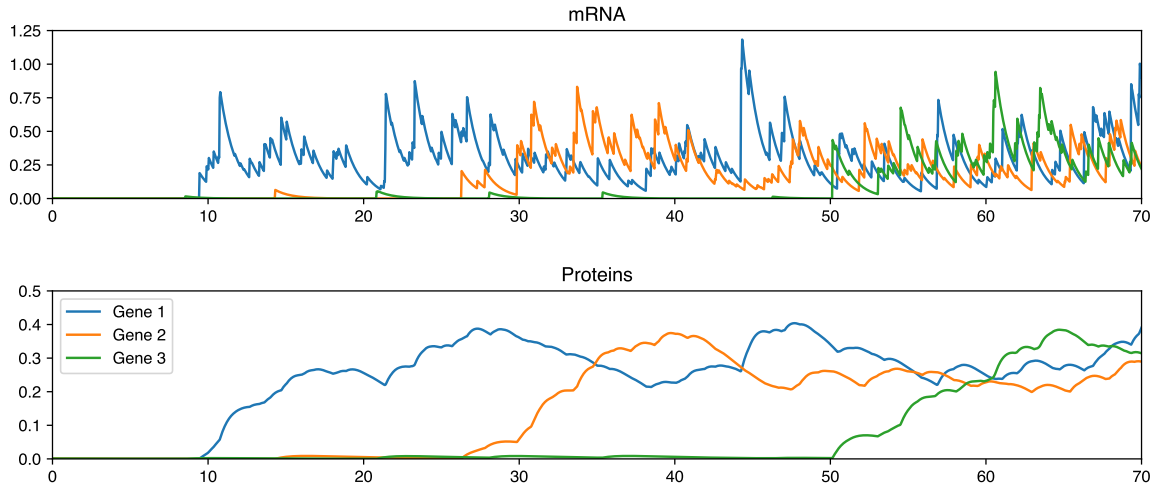


Figure 1.7 – Toujours le même modèle de réseau, mais cette fois en version *bursty*.

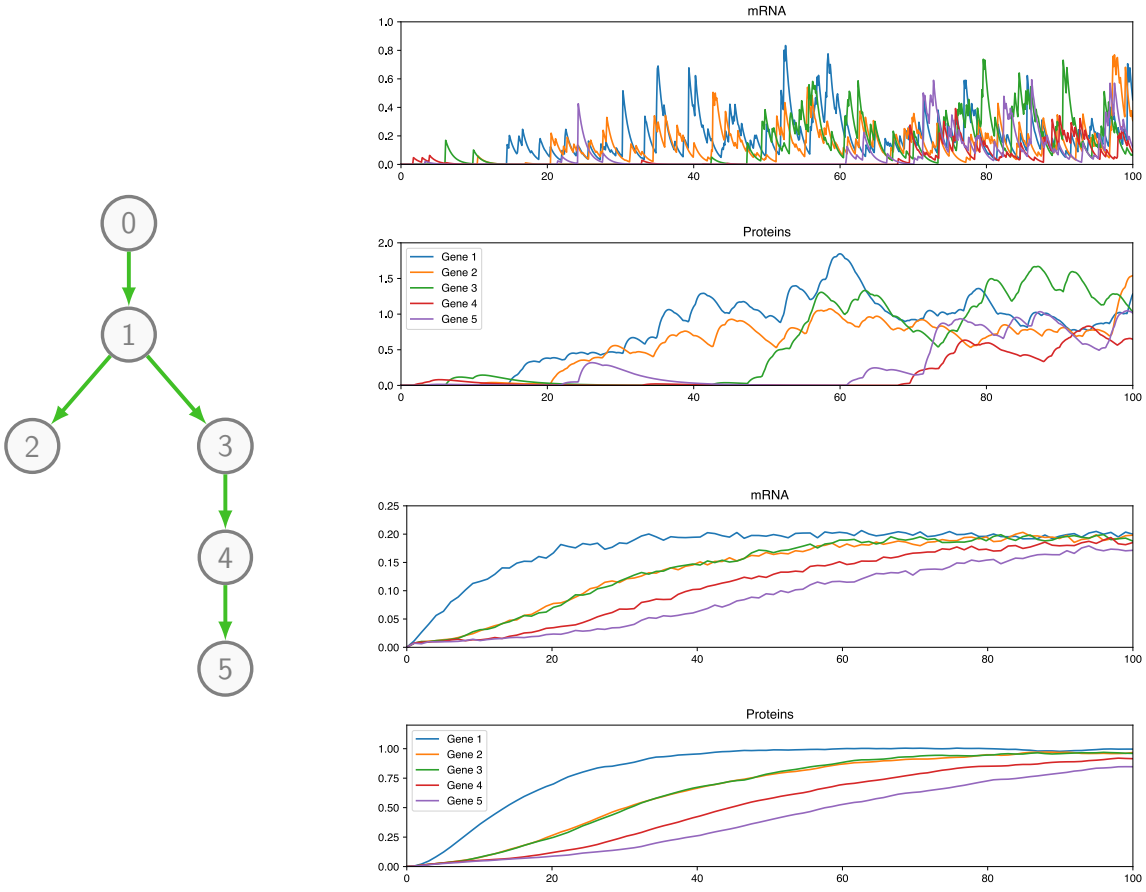


Figure 1.8 – Un réseau légèrement plus complexe. Noter que dans ce cas, les données de population (trajectoires moyennes estimées, en bas) ne permettent pas de distinguer les gènes 2 et 3.

Chapitre 2

Du modèle physique vers un modèle statistique

Nous avons à présent un modèle de réseau de régulation bien défini – par (1.5) ou sa version adimensionnée (1.14) – qui permet de simuler les trajectoires de cellules individuelles dans l'espace des niveaux de gènes. De plus, nous avons fait en sorte que les interactions soient paramétrées par une matrice $\theta = (\theta_{ij})_{1 \leqslant i,j \leqslant n}$ où θ_{ij} représente l'influence du gène j sur le gène i – forme d'interaction (1.11)-(1.12) – grâce à un modèle sous-jacent décrivant la chromatine. La question de l'inférence peut désormais se poser ainsi : étant donné un ensemble d'observations, peut-on trouver une valeur de θ qui permette de reproduire ces observations le plus fidèlement possible ?

Il faut d'abord préciser ce que l'on entend par *le plus fidèlement possible*. Profitant du fait que notre modèle physique correspond assez bien aux données sans avoir besoin de recourir à un quelconque « bruit de mesure », nous allons directement utiliser sa distribution [1] comme une *vraisemblance statistique*. Concrètement, nous allons supposer que les données de cellules uniques sont des échantillons indépendants de cette distribution : l'inférence se ramènera alors à un problème classique de maximisation de la vraisemblance (ou une estimation de la loi conditionnelle de θ sachant les données dans un cadre bayésien). L'indépendance des cellules est bien sûr une hypothèse forte, mais dans notre contexte *in vitro*, cela semble acceptable en première approximation.

La distribution qui nous intéresse est donc la solution stationnaire de l'équation maîtresse (1.17) que l'on ne sait pas résoudre dans le cas général.[2] En fait, on ne sait même pas résoudre le cas d'un gène isolé avec rétroaction sur lui-même puisque comme évoqué précédemment, la structure double « ARNm + protéines » est très difficile à traiter mathématiquement. Il s'agit par conséquent de trouver une approximation qui soit explicite sans être trop mauvaise. Dans ce chapitre, on commence par simplifier le modèle complet (1.14) en un modèle plus accessible dans lequel seuls les promoteurs et les protéines interviennent, puis on présente deux stratégies : la première est heuristique et aboutit à une pseudo-vraisemblance avec néanmoins des propriétés intéressantes, et la deuxième est basée sur la résolution exacte d'une classe de cas particuliers, fournissant une véritable vraisemblance assez pratique tout en semblant préserver la structure fondamentale de la distribution exacte.

1. Plus précisément, nous considérons ici sa loi invariante. Cela néglige à première vue la possibilité d'une évolution dans le temps qui a évidemment aussi son importance : l'object du chapitre 6 est précisément de prendre en compte l'aspect temporel, et nous en discuterons également dans le chapitre 7 et la conclusion.

2. Sans surprise car c'est un *many-body problem*, en d'autres termes un problème à n corps probabiliste.

Remarque 2.1. Dès que l'on parlera d'inférence basée sur la forme d'interaction (1.11)-(1.12), on se focalisera sur $\theta \in \mathcal{M}_n(\mathbb{R})$ en supposant que les hyperparamètres $k_{0,i}$, $k_{1,i}$, m_{ij} et s_{ij} sont connus. En pratique, nous fixerons s_{ij} à une valeur pertinente et nous introduirons une phase préliminaire d'estimation de $k_{0,i}$, $k_{1,i}$ et m_{ij}, basée sur le cas $\theta = 0$.

2.1 Simplification du modèle

On considère le processus $(E_t, Y_t, X_t)_{t\geqslant 0}$ correspondant au réseau adimensionné (1.14) et on se place dans le cas $d_1 < d_0$: autrement dit, la durée de vie moyenne d'une molécule d'ARNm est plus courte que celle de la protéine associée, ce qui est cohérent avec la réalité biologique (Schwanhäusser *et al.*, 2011). Cette séparation est intéressante du point de vue de la cellule car elle permet de convertir un signal *numérique* (bursts d'ARNm) en un signal *analogique* (niveaux de protéines). De manière plus pragmatique, elle permet de se ramener au modèle sans ARNm défini par

$$\forall i \in [\![1, n]\!], \quad \begin{cases} E_{t,i} : 0 \xrightarrow{a_i(X_t)} 1, \;\; 1 \xrightarrow{b_i(X_t)} 0 \\ \dot{X}_{t,i} = d_{1,i}(E_{t,i} - X_{t,i}) \end{cases} \tag{2.1}$$

puis de définir la loi conditionnelle de l'ARNm sachant les protéines comme le produit des lois stationnaires provenant du modèle de gène isolé (cf. section 1.1.4), c'est-à-dire :

$$\mathcal{L}(Y_t|X_t) = \bigotimes_{i=1}^{n} \text{Beta}\left(\frac{a_i(X_t)}{d_{0,i}}, \frac{b_i(X_t)}{d_{0,i}}\right). \tag{2.2}$$

Cette approximation peut sembler quelque peu brutale à première vue. En fait, l'idée est que comme les protéines sont plus lentes que l'ARNm ($d_1 < d_0$), ce sont elles qui déterminent la vitesse globale de l'évolution du réseau (on pourra s'en convaincre en regardant par exemple l'évolution des moyennes sur la Figure 1.8). Combiné avec le régime bursty ($b \gg a$ et $b \gg d_0$), on aboutit au fait que les niveaux d'ARNm fluctuent rapidement et ne sont que très peu corrélés directement à ceux des protéines, ce qui est vraiment observé en pratique (Albayrak *et al.*, 2016). Ainsi, les dépendances entre les niveaux d'ARNm vont venir non pas des dépendances entre les états des promoteurs eux-mêmes, mais plutôt des dépendances entre leurs paramètres $a_i(X_t)$ et $b_i(X_t)$, et a fortiori entre les protéines X_t. Ce phénomène sera illustré avec un exemple de réseau *toggle-switch* au chapitre 6, mais apportons à présent une justification un peu plus quantitative du modèle réduit (2.1).

2.1.1 Éléments de justification

Donnons déjà une première explication heuristique du fait que l'ARNm peut bien être enlevé du réseau de cette façon.[3] Pour cela, concentrons-nous sur un gène en abandonnant momentanément l'indice i. Soient $t_1 > t_0 \geqslant 0$ et $E \in \{0, 1\}$, et supposons que $E_t = E$ pour tout $t \in [t_0, t_1]$. Fixons $(Y_0, X_0) \in [0, 1] \times [0, 1]$. Lorsque $d_1 < d_0$, la solution du système différentiel linéaire :

$$\begin{cases} \dot{Y}_t = d_0(E_t - Y_t) \\ \dot{X}_t = d_1(Y_t - X_t) \end{cases}$$

3. Outre le fait que ce type de modèle est parfois introduit directement (Pájaro *et al.*, 2017).

qui préserve bien sûr $(Y_t, X_t) \in [0,1] \times [0,1]$, est donnée pour $t \in [t_0, t_1]$ par

$$\begin{cases} Y_t = E + (Y_0 - E)e^{-d_0(t-t_0)} \\ X_t = E + (X_0 - E)e^{-d_1(t-t_0)} + \dfrac{d_1}{d_0 - d_1}(Y_0 - E)\left(e^{-d_1(t-t_0)} - e^{-d_0(t-t_0)}\right). \end{cases} \tag{2.3}$$

Si $d_1 \ll d_0$, alors en utilisant le fait que $|Y_0 - E| \leqslant 1$ et $|e^{-d_1(t-t_0)} - e^{-d_0(t-t_0)}| \leqslant 1$, on obtient

$$X_t \approx E + (X_0 - E)e^{-d_1(t-t_0)}$$

et ainsi X_t approche la solution de l'équation différentielle $\dot{X}_t = d_1(E_t - X_t)$.

L'idée précédente fonctionne bien sûr uniquement sur les intervalles de temps où E_t est constant, et on voudrait s'assurer que l'erreur d'approximation n'explose pas sur une période plus longue, c'est-à-dire lorsque E_t varie dans le temps.[4] Nous allons cette fois donner un résultat rigoureux, mais valable uniquement pour un gène dont les taux de transition a et b du promoteur sont constants (et par conséquent isolé des autres gènes). Il devrait être possible de généraliser cela aux réseaux par une méthode de couplage, mais le cas du gène isolé suffit à donner l'intuition que le modèle simplifié (2.1) est pertinent.

Supposons donc a et b constantes, $d_1 < d_0$ et notons $\delta = d_1/(d_0 - d_1)$. Tout l'aléa est ainsi contenu dans $(E_t)_{t \geqslant 0}$, qui est un processus markovien de sauts à deux états. L'erreur d'approximation de X_t faite par (2.1) est aléatoire : on voudrait par exemple contrôler la moyenne de sa valeur absolue. Dans ce qui suit, on note $(T_k)_{k \geqslant 0}$ la suite des instants de saut avec $T_0 = 0$ par convention, et $(U_k)_{k \geqslant 1}$ la suite des temps d'attente entre les sauts (i.e. $U_k = T_k - T_{k-1}$). On considère finalement $R_k = |\widehat{X}_t - X_t| \in [0,1]$ l'erreur absolue à l'instant T_k, où $\widehat{X}_t$ est défini en remplaçant la deuxième ligne de (2.3) par $\widehat{X}_t = E + (\widehat{X}_0 - E)e^{-d_1(t-t_0)}$.

Proposition 2.2. *Il existe des constantes explicites $\alpha, r \in]0,1[$ telles que*

$$\mathbb{E}[R_k] \leqslant r^k \mathbb{E}[R_0] + (1 - r^k)\alpha, \quad \forall k \geqslant 0.$$

En particulier, si $R_0 = 0$ alors $\mathbb{E}[R_k] \leqslant \alpha$ pour tout $k \geqslant 0$.

Démonstration. En utilisant (2.3) et comme $|Y_t - E_t| \leqslant 1$ pour tout $t \geqslant 0$, on obtient

$$R_{k+1} \leqslant e^{-d_1 U_{k+1}} R_k + \delta\left(e^{-d_1 U_{k+1}} - e^{-d_0 U_{k+1}}\right), \quad \forall k \geqslant 0.$$

On peut facilement calculer l'espérance des deux cotés de cette inégalité, par indépendance des U_k et par construction de R_k. Comme les U_k sont distribués alternativement selon les lois exponentielles $\mathcal{E}(a)$ et $\mathcal{E}(b)$, on obtient après simplifications :

$$\mathbb{E}[R_{k+1}] \leqslant r\mathbb{E}[R_k] + c$$

pour tout $k \geqslant 0$, où les constantes r et c sont définies (de manière non optimale ici) par

$$r = \frac{\max\{a,b\}}{\max\{a,b\} + d_1} \quad \text{et} \quad c = d_1 \max\left\{\frac{a}{(a+d_0)(a+d_1)}, \frac{b}{(b+d_0)(b+d_1)}\right\}.$$

On en déduit le résultat par récurrence en posant $\alpha = c/(1-r)$. □

4. Elle ne risque pas d'exploser puisqu'elle est bornée par 1 (car $X_t \in [0,1]$ dans les deux cas). Mais on voudrait bien sûr qu'elle reste proche de 0 et n'ait pas tendance à augmenter au cours du temps.

À titre d'exemple, le cas $a = b$ fournit $\alpha = a/(a + d_0)$. On voit alors que l'erreur est d'autant plus faible que a est petit par rapport d_0 : cela correspond au régime où l'état de l'ARN ressemble beaucoup à celui du promoteur, et naturellement tout se passe alors comme si les protéines suivaient le modèle réduit (2.1). En pratique ce n'est pas tout à fait ce régime qui est observé mais plutôt $a < d_0 < b$, et la borne est largement améliorable en appliquant des majorations plus fines pour exploiter l'asymétrie entre a et b.

2.1.2 Équation maîtresse du modèle simplifié

On note toujours $\mathcal{E} = \{0, 1\}^n$ et $\Omega =]0, 1[^n$. L'ARNm ayant été retiré du modèle pour être placé à part dans (2.2), nous considérons à présent le processus $(E_t, X_t)_{t\geqslant 0}$ défini par (2.1), dont l'espace d'états est $\mathcal{S} = \mathcal{E} \times \Omega$. En notant $u : (t, x) \mapsto u(t, x) = (u_e(t, x))_{e\in\mathcal{E}}$ la distribution jointe de (E_t, X_t), on a ainsi la nouvelle équation maîtresse

$$\partial_t u + \sum_{i=1}^{n} \partial_{x_i}(F_i u) = \sum_{i=1}^{n} K_i u \tag{2.4}$$

où la différence avec (1.17) est que F_i dépend maintenant de x_i et de $d_{1,i}$, c'est-à-dire :

$$F_i(x_i) = I_2 \otimes \cdots \otimes \underbrace{F^{(i)}(x_i)}_{\text{rang } i} \otimes \cdots \otimes I_2, \qquad K_i(x) = I_2 \otimes \cdots \otimes \underbrace{K^{(i)}(x)}_{\text{rang } i} \otimes \cdots \otimes I_2,$$

$$F^{(i)}(x_i) = \begin{pmatrix} -d_{1,i}x_i & 0 \\ 0 & d_{1,i}(1 - x_i) \end{pmatrix} \quad \text{et} \quad K^{(i)}(x) = \begin{pmatrix} -a_i(x) & b_i(x) \\ a_i(x) & -b_i(x) \end{pmatrix}.$$

Par ailleurs, il est clair que la Proposition 1.6 s'applique encore au processus $(E_t, X_t)_{t\geqslant 0}$: dans toute la suite, on se place sous ses hypothèses et on définit

$$\underline{a} = \min\{a_i(x) \,|\, (i, x) \in [\![1, n]\!] \times \Omega\} \quad \text{et} \quad \underline{b} = \min\{b_i(x) \,|\, (i, x) \in [\![1, n]\!] \times \Omega\} \tag{2.5}$$

qui sont donc des constantes strictement positives. Par ergodicité du processus, on sait que l'équation (2.4) admet une unique solution stationnaire (i.e. telle que $\partial_t u = 0$) à constante multiplicative près : les deux stratégies présentées dans ce chapitre vont consister à trouver une fonction *simple* qui soit une approximation raisonnable de cette solution.

2.1.3 Moyennisation des promoteurs

Présentons maintenant un résultat connu, que nous n'utiliserons pas de manière directe mais qui constituera un point de comparaison avec notre approche. Comme évoqué à la fin du premier chapitre, on constate que lorsque les protéines sont bien plus lentes que les promoteurs ($d_1 \ll a, b$), elles ne sont plus influencées que par leurs états *moyens*. Il suffit alors de résoudre

$$\sum_{i=1}^{n} K_i(x)\overline{u}(x) = 0, \tag{2.6}$$

l'idée étant que quel que soit l'état $x \in \Omega$ des protéines, les promoteurs sont toujours dans le régime stationnaire associé à cet état : on parlera d'approximation *quasi-stationnaire* des promoteurs. On vérifie que le vecteur $\overline{u}(x) = \bigotimes_{i=1}^{n}(b_i(x), a_i(x))^\top$ est l'unique solution de (2.6) à constante multiplicative près, ce qui revient à dire que sous cette approximation,

les promoteurs sont indépendants sachant $X_t = x$ et tels que $\mathbb{E}[E_{t,i}|X_t = x] = p_i(x)$, où

$$p_i(x) = \frac{a_i(x)}{a_i(x) + b_i(x)}, \quad \forall i \in [\![1, n]\!]. \tag{2.7}$$

Finalement, la forme du modèle (2.1) suggère que la dynamique des protéines se rapproche de celle, déterministe, du système d'équations différentielles ordinaires

$$\dot{X}_{t,i} = d_{1,i} \left(\frac{a_i(X_t)}{a_i(X_t) + b_i(X_t)} - X_{t,i} \right), \quad \forall i \in [\![1, n]\!]. \tag{2.8}$$

On peut remarquer que même si les protéines suivent une dynamique linéaire à l'échelle des promoteurs d'après (2.1), on a manifestement par (2.8) une dynamique non linéaire à l'échelle du réseau de gènes.

Remarque 2.3. On dira que la distribution $\mathcal{L}(X_t|X_0 = x_0)$ se *concentre* autour de la valeur à l'instant t de l'unique solution de (2.8) vérifiant $X_0 = x_0$. Intuitivement, cela fait disparaître les fluctuations qui permettent aux cellules de passer d'un minimum énergétique local à un autre : sans changer les hypothèses qui assurent l'ergodicité du PDMP, le système dynamique (2.8) n'a plus aucune raison d'être ergodique (on peut avoir plusieurs points d'équilibre attractifs, par exemple dans le cas du *toggle switch*, cf. chapitre 6).

Il se trouve que le système (2.8) est bien justifié mathématiquement en tant que limite du processus $(X_t)_{t \geqslant 0}$. Une façon d'étudier cette limite est de modifier légèrement (2.1), en multipliant a_i et b_i par un paramètre $\rho > 0$ qui permet de quantifier simplement le rapport entre l'échelle de temps des promoteurs et celle des protéines :

$$\forall i \in [\![1, n]\!], \quad \begin{cases} E_{t,i} : 0 \xrightarrow{\rho a_i(X_t)} 1, \;\; 1 \xrightarrow{\rho b_i(X_t)} 0 \\ \dot{X}_{t,i} = d_{1,i}(E_{t,i} - X_{t,i}) \end{cases} \tag{2.9}$$

Cela revient à multiplier par ρ le membre de droite dans l'équation maîtresse (2.4), et on vérifie que (2.7) reste inchangé. On utilisera le paramètre ρ dans toute la suite de ce chapitre lorsqu'on voudra étudier ce régime. On a alors le résultat suivant, qui est un cas particulier de (Faggionato *et al.*, 2010, Theorem 2.1).

Proposition 2.4. *Soient $X_0 \in \Omega$ et $T \in \mathbb{R}_+$ fixés. Lorsque $\rho \to +\infty$, la trajectoire $(X_t)_{t \in [0,T]}$ converge en probabilité, uniformément sur $[0, T]$, vers la solution du système* (2.8).

On trouvera de nombreux détails ainsi que d'autres résultats en lien avec cette limite dans Faggionato *et al.* (2009). Il peut être très utile de simuler les trajectoires de (2.8) pour avoir une idée du comportement du réseau (points d'équilibre, etc.), mais en pratique la séparation d'échelles de temps n'est souvent pas assez grande pour que l'on puisse se passer du modèle stochastique : des expériences quantitatives comme celle de Schwanhäusser *et al.* (2011) suggèrent la valeur moyenne $\rho \approx 5$, pour laquelle passer à la limite $\rho \to +\infty$ semble assez risqué. Un compromis intéressant consiste à garder les petites fluctuations autour de la limite déterministe, i.e. considérer la *limite de diffusion* du modèle (2.9). Il se trouve que celle-ci est bien définie, comme cas particulier de Pakdaman *et al.* (2012), mais aussi qu'elle ne simplifie absolument pas notre étude en comparaison du formalisme PDMP. Nous en resterons donc au modèle réduit (2.1) – et sa version (2.9) avec ρ – qui semblent constituer le meilleur compromis en vue d'obtenir notre modèle statistique.

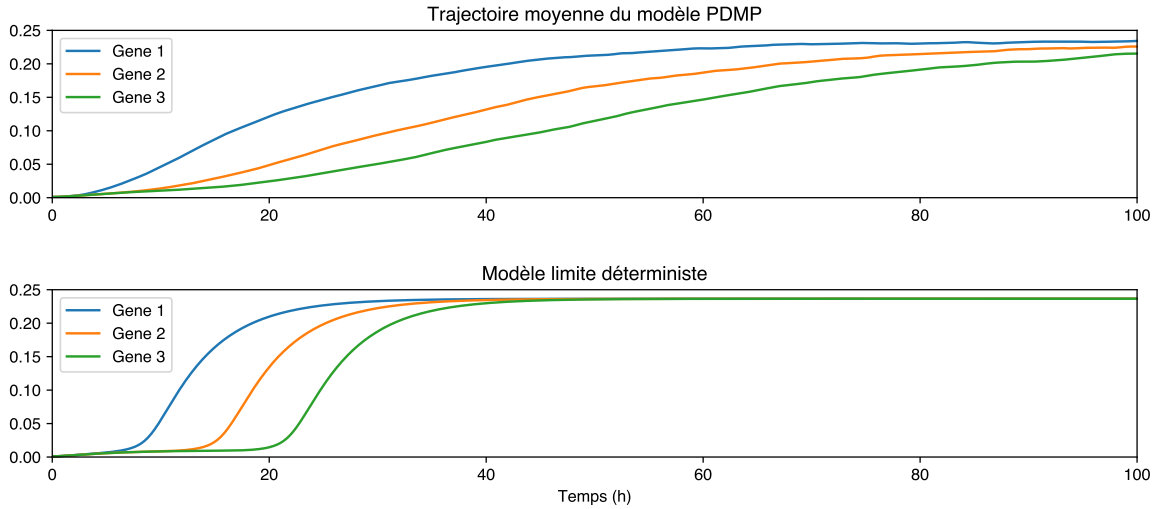


Figure 2.1 – Trajectoire moyenne des protéines suivant le modèle PDMP réduit (2.1), estimée avec 1000 cellules, comparée à la trajectoire du modèle limite déterministe (2.8). Les paramètres utilisés sont les mêmes que pour la Figure 1.5. S'il respecte bien l'ordre des gènes, le modèle déterministe représente manifestement assez mal la moyenne du modèle stochastique.

Remarquons finalement que si l'on choisit a_i et b_i à partir de l'exemple 1.3, le terme $p_i(x)$ défini par (2.7) prend la forme d'une fonction de Hill assez courante pour modéliser les réseaux de gènes (Mizeranschi *et al.*, 2015). Plus généralement, les modèles classiques que l'on peut trouver dans la littérature sont souvent des systèmes déterministes apparentés à (2.8), y compris ceux utilisés pour le *benchmark* de méthodes d'inférence (Marbach *et al.*, 2010). En particulier, ce type de modèle est utilisé pour simuler des données de population, c'est-à-dire les trajectoires moyennes des protéines.[5] Dans le cas de notre modèle PDMP et avec le rapport $d_0/d_1 = 5$, la Figure 2.1 montre que le modèle (2.8) est loin d'être satisfaisant.

2.2 Approche par pseudo-vraisemblance

Dans cette section, nous détaillons une méthode heuristique pour obtenir une fonction simple qui approche la solution stationnaire de l'équation maîtresse (2.4). L'idée de départ vient de physiciens qui ont eu l'idée remarquable de représenter le couple promoteur-protéine à la manière d'un modèle spin-boson en théorie quantique des champs (Sasai et Wolynes, 2003 ; Walczak *et al.*, 2005 ; Zhang et Wolynes, 2014). L'approximation considérée correspond alors à la « méthode de Hartree-Fock non restreinte » de la physique, que Walczak *et al.* (2005) baptisent *self-consistent proteomic field* (SCPF) dans ce contexte. Les auteurs obtiennent une approximation relativement simple mais définie de manière récursive, ce qui est très contraignant puisque notre but est de l'utiliser comme une vraisemblance statistique. En fait, la complexité vient essentiellement du fait qu'ils partent de la version discrète du modèle (2.1), c'est-à-dire définie comme le processus de sauts de départ (1.2). Nous verrons que la version PDMP permet d'avoir une forme totalement explicite et même simple, ce qui la rend bien plus intéressante pour parvenir à nos fins : pour la distinguer de la version discrète d'origine, nous l'appellerons simplement *approximation de Hartree*.

D'un point de vue statistique, cette approximation peut être considérée comme une

5. Ou même la moyenne de l'ARNm si ce dernier est utilisé comme *proxy* pour les protéines.

pseudo-vraisemblance dans le sens où il s'agit d'une fonction définie sur $\mathcal{S} = \mathcal{E} \times \Omega$ et à valeurs positives, mais qui n'est pas une densité de probabilité au sens strict puisque son intégrale sur $\mathcal{S}$ ne vaut pas toujours 1. Nous verrons par ailleurs dans la section 2.3 qu'il existe un lien entre l'approximation de Hartree et la pseudo-vraisemblance de Besag (1975) souvent utilisée en statistique spatiale. Dans la pratique, nous intégrerons la fonction sur les états des promoteurs de façon à obtenir une approximation de la densité stationnaire des protéines : en notant respectivement Y et X les niveaux d'ARNm et de protéines en régime stationnaire, puis en nous souvenant de l'hypothèse de séparation d'échelles de temps (2.2), nous aurons ainsi des fonctions $f_X(x)$ et $f_{Y|X}(y|x)$ *explicites* telles que

$$g(x, y) = f_{Y|X}(y|x) f_X(x)$$

soit une approximation de la densité du couple (X, Y). Comme seul l'ARNm Y est observé, cela correspond à un modèle à variables latentes assez typique (indépendance des variables observées Y_i sachant la variable cachée X).

L'application de cette méthode sera l'objet du chapitre 6, qui présente quelques résultats concrets ainsi que les détails techniques de l'inférence. Cette section est quant à elle consacrée aux aspects théoriques : nous commençons par présenter la construction de l'approximation de Hartree, puis nous abordons une propriété fort intéressante de cette dernière, liée à sa concentration lorsque $\rho \to +\infty$.

2.2.1 L'approximation de Hartree

L'idée consiste à scinder le problème de départ (2.4) de dimension 2^n en n problèmes indépendants de dimension 2, en « gelant » les x_j pour $j \neq i$ où i est fixé. On rassemble alors les solutions obtenues en prenant leur produit tensoriel pour produire une approximation de la vraie densité (Walczak *et al.*, 2005). Plus précisément, on obtient à partir de (2.4) un problème réduit pour chaque gène i :

$$\partial_t u^i + \partial_{x_i}(F^{(i)} u^i) = K^{(i)} u^i \tag{2.10}$$

où $u^i(t, x) = (u_0^i(t, x), u_1^i(t, x))^\top$ satisfait la condition initiale $u^i(0, x) = u^{i,0}(x)$, la condition aux bords $F^{(i)}(x_i) u^i(x) \to 0$ lorsque $x_i \to 0$ ou 1, et la condition de probabilité

$$\int_0^1 [u_0^i(t, x) + u_1^i(t, x)] dx_i = 1$$

pour tout $t \geqslant 0$ et $(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) \in]0, 1[^{n-1}$. Par conséquent, chaque u^i est une densité de probabilité par rapport à $(e_i, x_i) \in \{0, 1\} \times]0, 1[$, mais pas sur $\mathcal{E} \times \Omega$. Finalement, l'approximation de Hartree consiste à écrire

$$u(t, x) \approx \bigotimes_{i=1}^{n} u^i(t, x). \tag{2.11}$$

Noter que le symbole $\approx$ est par définition très flou. Néanmoins, on sait que l'égalité est vérifiée pour tout $t \geqslant 0$ lorsqu'elle est vraie à $t = 0$ et que pour tout $i \in [\![1, n]\!]$, a_i, b_i et $u^{i,0}$ ne dépendent que de x_i, c'est-à-dire lorsque les gènes sont indépendants.

Résolution du problème réduit. Il semble que la solution temporelle de (2.10) n'ait pas de forme explicite connue. En revanche, il est facile d'obtenir l'unique solution stationnaire dès que l'on connaît une primitive de

$$\lambda_i : x_i \mapsto \frac{a_i(x)}{d_{1,i}x_i} - \frac{b_i(x)}{d_{1,i}(1-x_i)}$$

qui n'est autre que la valeur propre non nulle de la matrice $M^{(i)} = K^{(i)}(F^{(i)})^{-1}$. En effet, en faisant le changement d'inconnue $v^i = F^{(i)}u^i$, l'équation stationnaire pour v^i obtenue à partir de (2.10) s'écrit

$$\partial_{x_i} v^i = M^{(i)} v^i$$

et il suffit alors, en exploitant de manière cruciale le fait que $M^{(i)}$ admet le vecteur propre constant $(-1,1)^\top$ associé à la valeur propre λ_i (l'autre valeur propre étant 0), de vérifier que $v^i = e^{\varphi_i}(-1,1)^\top$ est solution dès que $\partial_{x_i}\varphi_i = \lambda_i$. Si l'on connaît un tel φ_i, alors la solution stationnaire de (2.10) est donnée par

$$u_0^i(x) = {Z_i}^{-1} {x_i}^{-1} \exp(\varphi_i(x)) \quad \text{et} \quad u_1^i(x) = {Z_i}^{-1}(1-x_i)^{-1}\exp(\varphi_i(x)) \tag{2.12}$$

où Z_i est la constante de normalisation, qui peut dépendre de x_j pour $j \neq i$. Noter que l'existence des constantes $\underline{a} > 0$ et $\underline{b} > 0$ définies par (2.5) impose la limite 0 pour $\exp(\varphi_i(x))$ lorsque $x_i \to 0$ ou 1, de sorte que la condition aux bords est bien satisfaite. On obtient également les probabilités du promoteur E_i :

$$\mathbb{P}(E_i = 0) = \frac{Z_{0,i}}{Z_i} \quad \text{et} \quad \mathbb{P}(E_i = 1) = \frac{Z_{1,i}}{Z_i}$$

avec $Z_{0,i} = \int_0^1 x_i^{-1}\exp(\varphi_i(x))dx_i$, $Z_{1,i} = \int_0^1 (1-x_i)^{-1}\exp(\varphi_i(x))dx_i$ et $Z_i = Z_{0,i} + Z_{1,i}$.

Exemple 2.5. Lorsque a_i et b_i ne dépendent pas de x_i (i.e. pas d'auto-régulation), on obtient

$$\varphi_i(x) = \frac{a_i(x)}{d_{1,i}}\ln(x_i) + \frac{b_i(x)}{d_{1,i}}\ln(1-x_i),$$

ce qui fournit la solution classique

$$u_0^i(x) = \frac{\beta_i}{\alpha_i+\beta_i}\cdot\frac{x_i^{\alpha_i-1}(1-x_i)^{\beta_i}}{\mathrm{B}(\alpha_i,\beta_i+1)} \quad \text{et} \quad u_1^i(x) = \frac{\alpha_i}{\alpha_i+\beta_i}\cdot\frac{x_i^{\alpha_i}(1-x_i)^{\beta_i-1}}{\mathrm{B}(\alpha_i+1,\beta_i)} \tag{2.13}$$

où $\alpha_i = a_i(x)/d_{1,i}$ et $\beta_i = b_i(x)/d_{1,i}$. Cette forme permet de distinguer les probabilités du promoteur E_i et les lois conditionnelles de la protéine X_i sachant $E_i = 0$ ou 1, qui sont toutes les deux des lois Beta. Par ailleurs, comme E_i n'est en général pas observé, on considère souvent la loi marginale de X_i, qui est encore une loi Beta :

$$\underline{u}^i(x) = u_0^i(x) + u_1^i(x) = \frac{x_i^{\alpha_i-1}(1-x_i)^{\beta_i-1}}{\mathrm{B}(\alpha_i,\beta_i)}. \tag{2.14}$$

On peut constater que loi (2.2) de l'ARNm sachant les protéines est aussi de la forme (2.14), puisque l'équation (1.17) a la même forme en y que (2.4) en x. La différence est l'argument utilisé : dans le cas des protéines c'est l'approximation de Hartree, tandis que pour l'ARNm c'est un argument de quasi-stationnarité similaire à (2.6) pour les promoteurs.

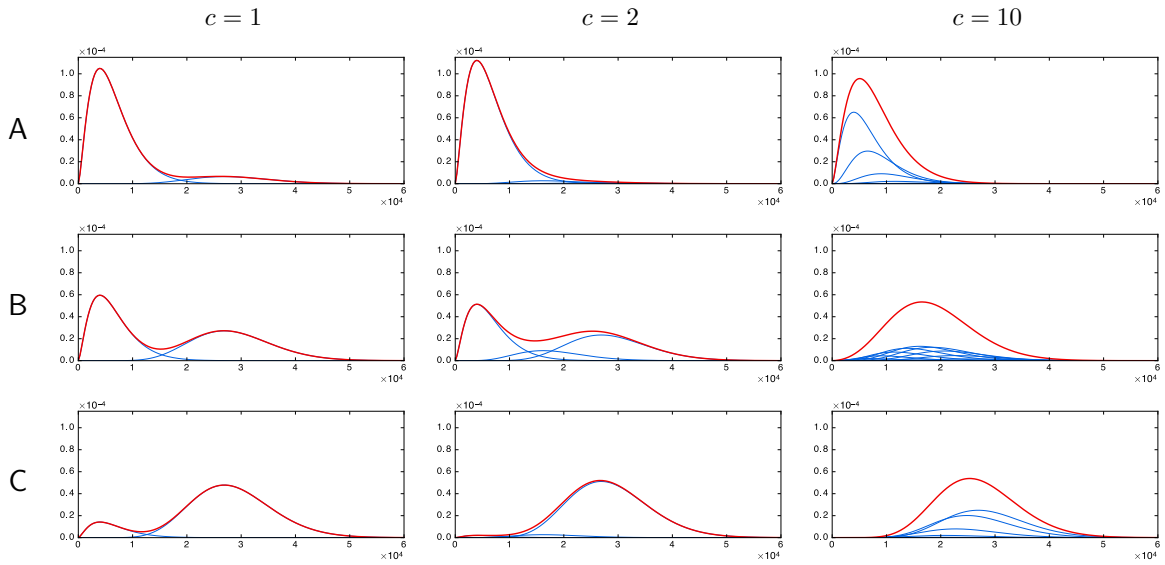


Figure 2.2 – Loi stationnaire des protéines (en rouge) et lois Beta sous-jacentes (en bleu) pour le modèle d'auto-activation (2.15) avec différentes valeurs de ν et c. Chaque colonne correspond à une valeur de c et chaque ligne correspond à une valeur de ν. (A) $\nu = \exp(-2)$, (B) $\nu = \exp(0) = 1$, (C) $\nu = \exp(2)$. La distribution est fortement bimodale pour les petites valeurs de c, tandis que les grandes valeurs rendent la distribution proche du cas unimodal sans auto-régulation (a constant).

Exemple 2.6. Il est aussi possible d'obtenir la loi stationnaire pour un gène avec auto-activation de la forme (1.11)-(1.12) issue de notre modèle de chromatine (section 1.2). En laissant tomber l'indice i pour simplifier, il s'agit du cas où b est constante et

$$a(x) = \frac{k_0 + k_1 \nu x^m}{1 + \nu x^m} \tag{2.15}$$

où ν rassemble tout ce qui ne dépend pas de x (i.e. x_i dans le réseau). En effet, on est dans le cas de la solution explicite (2.12) et en posant $c = (k_1 - k_0)/(md_1) > 0$, on en déduit que la distribution de la protéine en question est

$$\underline{u}(x) = Z^{-1} x^{-1} \left(x^{k_0/(d_1 c)} + \nu x^{k_1/(d_1 c)} \right)^c (1-x)^{b/d_1 - 1}. \tag{2.16}$$

Pour calculer Z, on peut se placer dans le cas $c \in \mathbb{N}^*$. En développant (2.16), on obtient alors $Z = \sum_{r=0}^{c} \binom{c}{r} \mathrm{B}(\alpha_r, \beta) \nu^r$ où $\alpha_r = ((c-r)k_0 + rk_1)/(d_1 c)$ et $\beta = b/d_1$. On a même une représentation de $\underline{u}$ comme un mélange de lois Beta :

$$\underline{u}(x) = \sum_{r=0}^{c} p_r f_r(x) \tag{2.17}$$

où

$$f_r(x) = \frac{x^{\alpha_r - 1}(1-x)^{\beta - 1}}{\mathrm{B}(\alpha_r, \beta)} \quad \text{et} \quad p_r = Z^{-1} \binom{c}{r} \mathrm{B}(\alpha_r, \beta) \nu^r.$$

La Figure 2.2 montre quelques exemples de la distribution (2.17), qui peut être bimodale ou pas en fonction de c, ou de manière équivalente m lorsque les autres paramètres sont fixés.

Remarque 2.7. Les taux de dégradation $d_{1,i}$ s'agrègent systématiquement à a_i et b_i en régime stationnaire, formant les termes $\alpha_i = a_i/d_{1,i}$ et $\beta_i = b_i/d_{1,i}$. Dans la suite de cette section, nous ferons l'abus de notation consistant à écrire a_i et b_i au lieu de α_i et β_i.

Distribution des protéines

La forme de la solution du problème réduit (2.12) fait qu'il est toujours possible d'intégrer l'approximation de Hartree (2.11) en régime stationnaire sur les états des promoteurs, même lorsqu'il y a de l'auto-régulation. Plus précisément, on remarque que

$$\sum_{e\in\mathcal{E}} \left[\bigotimes_{i=1}^{n} u^i(x)\right]_e = \sum_{e\in\mathcal{E}} \prod_{i=1}^{n} \frac{\exp(\varphi_i(x))}{Z_i(x)|e_i - x_i|} = \prod_{i=1}^{n} \frac{\exp(\varphi_i(x))}{Z_i(x)x_i(1-x_i)}$$

où l'on a rappelé la dépendance possible de Z_i par rapport aux x_j pour $j \neq i$. Ainsi, lorsque φ_i et Z_i sont des fonctions connues, on obtient une approximation *explicite* de la distribution jointe des protéines. Dans toute la suite, on considèrera directement cette version intégrée. On se basera sur la version avec promoteurs rapides (2.9), sachant qu'il suffit de prendre $\rho = 1$ pour retomber sur le modèle de départ (2.1).

Définition 2.8. L'approximation de Hartree (stationnaire) du modèle (2.9) est la fonction $\underline{h}_\rho$ définie par

$$\underline{h}_\rho(x) = \prod_{i=1}^{n} \frac{\exp(\rho\varphi_i(x))}{Z_i(x)x_i(1-x_i)}, \quad \forall x \in \Omega \tag{2.18}$$

où φ_i est définie comme dans (2.12) et $Z_i(x) = \int_0^1 {x_i}^{-1}(1-x_i)^{-1}\exp(\rho\varphi_i(x))dx_i$. On note $\underline{h} = \underline{h}_1$ l'approximation de Hartree correspondant à (2.1).

Finalement, on peut facilement construire l'approximation de Hartree d'un modèle de réseau à partir de la formule générale (2.18) en utilisant, pour chaque gène i, soit (2.14) si le gène n'a pas d'auto-régulation, soit (2.17) s'il est auto-activé avec la forme (2.15). Par ailleurs, quitte à augmenter encore l'erreur d'approximation, il est assez naturel de vouloir utiliser la forme (2.14) dans le cas général, c'est-à-dire même en présence d'auto-régulation. On va aussi considérer cette forme en tant qu'approximation généralisée.

Définition 2.9. L'approximation de Hartree généralisée du modèle (2.9) est la fonction h_ρ définie par

$$h_\rho(x) = \prod_{i=1}^{n} \frac{{x_i}^{\rho a_i(x)-1}(1-x_i)^{\rho b_i(x)-1}}{\mathrm{B}(\rho a_i(x), \rho b_i(x))}, \quad \forall x \in \Omega. \tag{2.19}$$

On note $h = h_1$ la version correspondant à (2.1).

Dans le cas des gènes indépendants sans auto-régulation (i.e. a_i et b_i constantes), on a $h(x) = \underline{h}(x) = \sum_{e\in\mathcal{E}} u_e(x)$ où u est la solution stationnaire exacte de (2.4). Intuitivement, $\underline{h}_\rho$ est plus précise que h_ρ si certains gènes présentent une auto-régulation, mais ici nous allons nous baser sur h_ρ qui a l'avantage d'avoir une forme générale simple et fixée. En outre, on peut remarquer deux propriétés intéressantes de h_ρ (et a fortiori de $\underline{h}_\rho$) :

- c'est bien une fonction positive et intégrable sur Ω ;
- on vérifie facilement que son intégrale sur Ω vaut 1 lorsqu'il n'y a pas d'auto-régulation et que le graphe des interactions est un *graphe orienté acyclique* (DAG).

Noter que l'intégrale ne vaut pas 1 en général et que même dans le cas d'un DAG sans auto-régulation, ce n'est manifestement pas la solution stationnaire exacte de (2.4).

2.2.2 Un résultat de concentration

Nous avons introduit une fonction $\underline{h}_\rho$ autoproclamée « approximation de Hartree » et sa généralisation h_ρ, mais pour le moment rien ne justifie leur titre d'*approximation* : on sait que $\underline{h}_\rho$ est bien la densité exacte lorsque a_i et b_i ne dépendent que de x_i, mais dans le cas contraire rien ne nous dit que $\underline{h}_\rho$ et h_ρ sont « proches » de la densité exacte. Dans cette section, nous obtenons un résultat partiel mais intéressant qui peut se résumer ainsi : lorsque $\rho \to +\infty$, la masse représentée par h_ρ se concentre autour de points particuliers qui sont exactement les états d'équilibre du système limite (2.8). C'est cohérent avec la Proposition 2.4, puisqu'on s'attend ainsi à ce que les modes éventuels de l'approximation de Hartree soient localisés aux mêmes endroits que la vraie densité. Ainsi, même si h_ρ n'est pas une approximation au sens strict où l'on disposerait d'une borne de l'erreur, ce résultat est encourageant pour utiliser h_ρ (ou $\underline{h}_\rho$ qui est plus précise) comme une pseudo-vraisemblance.

L'outil principal que l'on va utiliser dans cette section est la méthode de Laplace, que l'on rappelle brièvement. Le principe est de trouver un équivalent simple lorsque $\rho \to +\infty$ d'une intégrale du type

$$\int_I f(x) e^{-\rho g(x)} dx$$

où I est un ouvert de $\mathbb{R}$, en supposant qu'il existe un unique $x_0 \in I$ qui minimise g, et de plus que g est deux fois dérivable sur I avec $g''(x_0) > 0$. On a alors nécessairement $g'(x_0) = 0$, et un développement de Taylor à l'ordre 2 nous suggère qu'au voisinage de x_0,

$$g(x) \approx g(x_0) + \frac{1}{2} g''(x_0)(x - x_0)^2$$

et donc que l'on peut approcher notre intégrale par une version « gaussienne » qui va se concentrer autour du point x_0 quand $\rho \to +\infty$. Il s'avère que l'intuition fonctionne très bien puisque l'on peut montrer que

$$\int_I f(x) e^{-\rho g(x)} dx \underset{\rho \to +\infty}{\sim} \sqrt{\frac{2\pi}{g''(x_0)\rho}} f(x_0) e^{-\rho g(x_0)} \tag{2.20}$$

dès lors que $f(x_0) \neq 0$. La méthode se généralise très bien en dimension n : si I est maintenant un ouvert de $\mathbb{R}^n$ et $\det(H_g(x_0)) > 0$ où H_g est la matrice hessienne de g, on obtient

$$\int_I f(x) e^{-\rho g(x)} dx \underset{\rho \to +\infty}{\sim} \left(\frac{2\pi}{\rho}\right)^{\frac{n}{2}} \det(H_g(x_0))^{-\frac{1}{2}} f(x_0) e^{-\rho g(x_0)}. \tag{2.21}$$

Si l'on considère à présent la fonction h_ρ définie par (2.19), l'idée générale est d'appliquer une première fois la méthode de Laplace univariée (2.20) sur les intégrales au dénominateur (des fonctions Beta), ce qui fait apparaître la forme $\alpha(x)\exp(-\rho g(x))$ où l'on connaît les minima locaux de g, mais aussi un facteur « gaussien » $(2\pi/\rho)^{\frac{n}{2}}$. On peut alors appliquer la version multivariée (2.21) sur cette nouvelle fonction, ce qui a pour effet de simplifier le facteur $(2\pi/\rho)^{\frac{n}{2}}$ et fournit une limite finie explicite.

De manière surprenante, il se trouve que le résultat de concentration fait apparaître la divergence de Kullback–Leibler ou *entropie relative* entre certaines mesures de probabilité.

On aura seulement besoin de la définition dans le cas fini que l'on rappelle ci-dessous.

Définition 2.10. Soient μ et ν deux mesures de probabilité sur un ensemble fini $\mathcal{S}$. La *divergence de Kullback–Leibler* de μ par rapport à ν est la quantité $\mathrm{D}(\mu\|\nu)$ définie par

$$\mathrm{D}(\mu\|\nu) = \sum_{x\in\mathcal{S}} \mu(x) \ln\left(\frac{\mu(x)}{\nu(x)}\right)$$

sous réserve que $\mu \ll \nu$ (i.e. pour tout $x \in \mathcal{S}$, $\nu(x) = 0 \Rightarrow \mu(x) = 0$).

L'appellation « divergence » provient du fait fondamental que $\mathrm{D}(\mu\|\nu) \geqslant 0$ avec égalité si et seulement si $\mu = \nu$, conséquence immédiate de l'*inégalité de Pinsker* :

$$\mathrm{D}(\mu\|\nu) \geqslant \frac{1}{2}\|\mu - \nu\|_1^2$$

où $\|\mu - \nu\|_1 = \sum_{x\in\mathcal{S}} |\mu(x) - \nu(x)|$. Pour comprendre l'appellation « entropie relative », on peut faire le lien avec l'entropie de Shannon classique.[6] Pour toute mesure de probabilité μ sur $\mathcal{S}$, on a en effet

$$\mathrm{Ent}(\mu) = \mathrm{Ent}(\overline{\nu}) - \mathrm{D}(\mu\|\overline{\nu})$$

où $\overline{\nu}$ est la loi uniforme sur $\mathcal{S}$. Ainsi, augmenter l'entropie de μ correspond à diminuer sa divergence de Kullback–Leibler par rapport à la mesure de référence $\overline{\nu}$, qui est justement celle d'entropie maximale.

Il sera également pratique, afin d'éviter un recours intempestif aux ε, d'avoir une notion d'équivalence uniforme entre deux familles $(f_\rho)_{\rho\geqslant 0}$ et $(g_\rho)_{\rho\geqslant 0}$ de fonctions $\Omega \to \mathbb{R}^*$ indexées par $\rho \in \mathbb{R}_+$: étant donné $A \subset \Omega$, on dira qu'elles sont *uniformément équivalentes sur* A si $\sup_{x\in A} |f_\rho(x)/g_\rho(x) - 1|$ tend vers 0 lorsque $\rho \to +\infty$, et on notera dans ce cas

$$f_\rho \underset{\rho\to+\infty}{\overset{A}{\sim}} g_\rho$$

en remarquant que comme pour les équivalents classiques, la relation est stable par passage à l'inverse et les équivalents uniformes peuvent être multipliés entre eux.

Proposition 2.11. *L'approximation de Hartree généralisée h_ρ définie par* (2.19) *vérifie*

$$h_\rho \underset{\rho\to+\infty}{\overset{\Omega}{\sim}} \left(\frac{\rho}{2\pi}\right)^{\frac{n}{2}} f\exp(-\rho g) \tag{2.22}$$

avec f et g définies par

$$f(x) = \prod_{i=1}^{n} \frac{1}{x_i(1-x_i)}\sqrt{\frac{a_i(x)b_i(x)}{a_i(x)+b_i(x)}} \quad \textit{et} \quad g(x) = \sum_{i=1}^{n}(a_i(x)+b_i(x))\,\mathrm{D}(\mu_i(x)\|\nu_i(x))$$

où $\nu_i(x)$ et $\mu_i(x)$ sont les lois de Bernoulli de paramètres respectifs x_i et $p_i(x)$ défini par (2.7).

Démonstration. Partant de la définition, il s'agit de trouver un équivalent uniforme sur Ω de la fonction

$$\prod_{i=1}^{n} \frac{1}{\mathrm{B}(\rho a_i, \rho b_i)} = \prod_{i=1}^{n} \frac{\Gamma(\rho a_i + \rho b_i)}{\Gamma(\rho a_i)\Gamma(\rho b_i)}.$$

6. On gardera la base naturelle du logarithme en posant ici $\mathrm{Ent}(\mu) = -\sum_{x\in\mathcal{S}} \mu(x)\ln(\mu(x))$.

Nous allons simplement appliquer la méthode de Laplace $3n$ fois à la fonction Γ, la seule subtilité étant que l'on veut des équivalents uniformes sur Ω. Tout d'abord, la formule (2.20) fournit

$$\Gamma(z) \underset{z\to+\infty}{\sim} \sqrt{2\pi} z^{z-\frac{1}{2}} e^{-z}$$

ou autrement dit,

$$\forall \varepsilon > 0,\ \exists M_\varepsilon \in \mathbb{R}_+^*,\ \forall z \geqslant M_\varepsilon, \quad \left| \frac{\Gamma(z)}{\sqrt{2\pi} z^{z-\frac{1}{2}} e^{-z}} - 1 \right| \leqslant \varepsilon.$$

Soit maintenant $\varepsilon > 0$ fixé. On considère un tel M_ε et on pose $\rho_\varepsilon = M_\varepsilon / \min(\underline{a}, \underline{b})$ où $\underline{a}$ et $\underline{b}$ sont définis par (2.5) et sont bien des quantités strictement positives par hypothèse sur a_i et b_i. Pour $i \in [\![1, n]\!]$, on a alors d'après l'inégalité précédente :

$$\forall \rho \geqslant \rho_\varepsilon, \quad \forall x \in \Omega, \quad \left| \frac{\Gamma(\rho a_i(x))}{\sqrt{2\pi} (\rho a_i(x))^{\rho a_i(x) - \frac{1}{2}} e^{-\rho a_i(x)}} - 1 \right| \leqslant \varepsilon$$

et ainsi

$$\Gamma(\rho a_i) \underset{\rho\to+\infty}{\overset{\Omega}{\sim}} \sqrt{2\pi} (\rho a_i)^{\rho a_i - \frac{1}{2}} e^{-\rho a_i}.$$

On obtient de la même façon les équivalents analogues pour $\Gamma(\rho b_i)$ et $\Gamma(\rho a_i + \rho b_i)$, ce qui nous donne après simplifications :

$$\prod_{i=1}^n \frac{1}{\mathrm{B}(\rho a_i, \rho b_i)} \underset{\rho\to+\infty}{\overset{\Omega}{\sim}} \left(\frac{\rho}{2\pi}\right)^{\frac{n}{2}} \prod_{i=1}^n \sqrt{\frac{a_i b_i}{a_i + b_i}} \left(\frac{(a_i + b_i)^{a_i + b_i}}{{a_i}^{a_i} {b_i}^{b_i}} \right)^\rho$$

En posant $e_i(x) = x_i$ et après un dernier calcul, on a finalement

$$\prod_{i=1}^n \frac{{e_i}^{\rho a_i} (1 - e_i)^{\rho b_i}}{\mathrm{B}(\rho a_i, \rho b_i)} \underset{\rho\to+\infty}{\overset{\Omega}{\sim}} \left(\frac{\rho}{2\pi}\right)^{\frac{n}{2}} \times f \times \exp\left(-\rho \sum_{i=1}^n (a_i + b_i) R_i \right)$$

avec

$$R_i = p_i \ln\left(\frac{p_i}{e_i}\right) + (1 - p_i) \left(\frac{1 - p_i}{1 - e_i}\right) = \mathrm{D}(\mu_i \| \nu_i)$$

en se souvenant que $p_i = a_i/(a_i + b_i)$, d'où le résultat. □

Si la fonction f de la Proposition 2.11 n'est pas très informative, g l'est bien plus. En effet, lorsque ρ est grand, l'équivalent (2.22) indique que la fonction h_ρ aura tendance à se concentrer aux points qui minimisent g du fait de la forme exponentielle. Or, d'après la propriété fondamentale de la divergence de Kullback–Leibler et puisque $a_i + b_i > 0$ sur Ω, on sait que g est positive et s'annule exactement aux points $x \in \Omega$ qui annulent toutes les divergences $\mathrm{D}(\mu_i(x) \| \nu_i(x))$, c'est-à-dire tels que $\mu_i(x) = \nu_i(x)$, ou encore tels que

$$\forall i \in [\![1, n]\!], \quad p_i(x) = x_i.$$

On reconnaît précisément les points fixes du système déterministe (2.8), ce qui est plutôt rassurant : cela suggère que malgré sa construction purement empirique, l'approximation de Hartree n'est pas si mauvaise puisqu'elle se concentre aux bons endroits ! En ajoutant une hypothèse de régularité sur a_i et b_i, on obtient le résultat suivant qui précise la concentration dans les cas non dégénérés. Dans ce qui suit, on note $\Phi(x) = (p_1(x), \ldots, p_n(x))$, ce qui définit

bien une application $\Phi : \Omega \to \Omega$, continue par hypothèse sur les a_i et b_i.

Corollaire 2.12. *Soit $\mathcal{F} = \{x \in \Omega \,|\, \Phi(x) = x\}$ l'ensemble des points fixes du système (2.8). On suppose que $\mathcal{F}$ est fini, que a_i et b_i sont de classe $\mathcal{C}^2$ sur Ω pour tout $i \in [\![1, n]\!]$, et que $\det(H_g(x)) \neq 0$ pour tout $x \in \mathcal{F}$ où $H_g = (\partial_{x_i}\partial_{x_j} g)_{1 \leqslant i,j \leqslant n}$ est la matrice hessienne de g. Alors h_ρ converge en loi vers une somme de mesures de Dirac pondérées :*

$$h_\rho \xrightarrow[\rho \to +\infty]{\mathcal{L}} \sum_{\overline{x} \in \mathcal{F}} w(\overline{x}) \delta_{\overline{x}} \tag{2.23}$$

où les poids sont donnés par $w(\overline{x}) = f(\overline{x}) \det(H_g(\overline{x}))^{-\frac{1}{2}}$.

Démonstration. Remarquons tout d'abord que $\mathcal{F}$ contient au moins un élément d'après le théorème de Brouwer, et que comme g s'annule aux points de $\mathcal{F}$ et est strictement positive en dehors de $\mathcal{F}$ qui est supposé fini, on a nécessairement $\det(H_g(x)) \geqslant 0$ pour tout $x \in \mathcal{F}$. Il s'agit maintenant de montrer que pour toute fonction $\phi : \Omega \to \mathbb{R}$ continue et bornée,

$$\int_\Omega \phi(x) h_\rho(x) dx \xrightarrow[\rho \to +\infty]{} \sum_{\overline{x} \in \mathcal{F}} w(\overline{x}) \phi(\overline{x}).$$

Commençons par découper l'intégrale en

$$\int_\Omega \phi(x) h_\rho(x) dx = \int_{\Omega^+} \phi(x) h_\rho(x) dx \; + \int_{\Omega^-} \phi(x) h_\rho(x) dx$$

où $\Omega^+ = \{x \in \Omega \,|\, \phi(x) > 0\}$ et $\Omega^- = \{x \in \Omega \,|\, \phi(x) < 0\}$, qui sont bien Lebesgue-mesurables et même ouverts puisque ϕ est continue. Deux cas se présentent.

Si Ω^- est de mesure nulle, alors c'est l'ensemble vide et on sait que ϕh_ρ est strictement positive sur Ω. Dans ce cas, on vérifie facilement que l'équivalent uniforme (2.22) implique l'équivalent au sens classique

$$\int_\Omega \phi(x) h_\rho(x) dx \underset{\rho \to +\infty}{\sim} \left(\frac{\rho}{2\pi}\right)^{\frac{n}{2}} \int_\Omega \phi(x) f(x) \exp(-\rho g(x)) dx$$

et il n'y a plus qu'à appliquer la méthode de Laplace (2.21) à l'intégrale de droite pour obtenir le résultat (pour être sous l'hypothèse classique d'un unique point qui minimise g, on peut d'abord découper l'intégrale selon une partition finie de Ω qui sépare les points $\overline{x} \in \mathcal{F}$, puis sommer en remarquant que les $w(\overline{x})\phi(\overline{x})$ sont strictement positifs).

Sinon, on peut appliquer le même raisonnement en intégrant cette fois sur Ω^-, puisque dans ce cas $-\phi h_\rho$ est strictement positive sur Ω^-. On obtient

$$\int_{\Omega^-} f(x) h_\rho(x) dx \xrightarrow[\rho \to +\infty]{} \sum_{\overline{x} \in \mathcal{F} \cap \Omega^-} w(\overline{x}) f(\overline{x}).$$

Dans le cas où $\Omega^- = \Omega$ on a fini, et sinon il suffit de faire la même chose sur Ω^+ (qui est alors de mesure non nulle) puis de sommer pour obtenir le résultat. □

On trouvera une illustration de ce résultat dans le cas du *toggle-switch* au chapitre 6. Il convient néanmoins de remarquer un problème potentiel : la somme (2.23) porte sur *tous* les points fixes du système (2.8), et pas seulement les points fixes attractifs. Lorsque $\rho \to +\infty$, il y a donc de la masse qui s'accumule aux équilibres instables, ce qui est aberrant d'un

point de vue physique. Il s'agit bien d'une faiblesse de l'approximation de Hartree qui est vérifiable numériquement. En pratique, on est toujours assez loin de la limite $\rho \to +\infty$ et la répartition de la masse de h_ρ semble assez cohérente en général.

Exemple 2.13. Dans le cas où a_i et b_i sont des constantes, il est facile de voir que $\mathcal{F} = \{\overline{x}\}$ où $\overline{x}_i = a_i/(a_i + b_i)$ pour $i \in [\![1, n]\!]$. On obtient alors

$$\det(H_g(\overline{x})) = \prod_{i=1}^{n} \frac{(a_i + b_i)^3}{a_i b_i}$$

et en appliquant le corollaire 2.12, on retrouve bien la convergence $h_\rho \to \delta_{\overline{x}}$. Noter que dans ce cas, h_ρ est en fait la loi stationnaire exacte du processus $(X_t)_{t \geqslant 0}$, et on retrouve donc le résultat que l'on aurait obtenu en passant par la Proposition 2.4.

2.3 Approche par résolution explicite

L'approximation de Hartree $\underline{h}$ est assez prometteuse pour servir de méthode d'inférence, mais elle ne fournit pas un modèle statistique rigoureux dans le sens où il s'agit seulement d'une pseudo-vraisemblance en général. En particulier, on n'a a priori aucun moyen théorique de s'assurer de l'identifiabilité de $\underline{h}$ par rapport à θ. Dans cette section, nous abordons une stratégie alternative basée sur des choix particuliers de fonctions a_i et b_i, constituant des classes de réseaux dont la loi stationnaire peut être calculée explicitement : l'idée est que si cette loi possède une certaine structure fondamentale, on pourra également l'utiliser en tant qu'approximation dans des cas que l'on ne sait pas résoudre mais qui semblent contenir la même structure fondamentale.

2.3.1 Une classe générale de solutions explicites

Commençons par donner un résultat simple mais très pratique, caractérisant toute une classe de fonctions d'interaction : celles-ci font directement apparaître une forme de potentiel énergétique qui se retrouve dans la loi stationnaire exacte.

Proposition 2.14. *On suppose qu'il existe une fonction $V : \Omega \to \mathbb{R}$ de classe $\mathcal{C}^1$ qui vérifie*

$$\forall x \in \Omega, \quad \frac{a_i(x)}{d_{1,i} x_i} - \frac{b_i(x)}{d_{1,i}(1 - x_i)} = -\frac{\partial V}{\partial x_i} \tag{2.24}$$

pour tout $i \in [\![1, n]\!]$. Alors la distribution stationnaire du réseau s'écrit, pour tout $(e, x) \in \mathcal{S}$:

$$u_e(x) = \frac{\exp(-V(x))}{A \prod_{i=1}^{n} |e_i - x_i|}$$

et en particulier la densité des protéines est donnée par

$$\underline{u}(x) = \frac{\exp(-V(x))}{A \prod_{i=1}^{n} x_i(1 - x_i)} \tag{2.25}$$

où $A = \int_\Omega \exp(-V(x)) \prod_{i=1}^{n} {x_i}^{-1}(1 - x_i)^{-1} dx$ est la constante de normalisation.

Démonstration. En faisant le changement d'inconnue $v = F_1 \cdots F_n u$ dans l'équation (2.4) en régime stationnaire, on remarque que celle-ci peut se réécrire

$$\sum_{i=1}^{n} F_i \partial_{x_i} v = \sum_{i=1}^{n} F_i K_i {F_i}^{-1} v$$

où l'on a utilisé de manière cruciale des commutations faciles à déduire de l'écriture tensorielle de F_i et K_i. On va encore exploiter l'existence d'un vecteur propre constant en x. En effet, sous les hypothèses de la proposition, si l'on pose $v(x) = \exp(-V(x)) \bigotimes_{i=1}^{n} (-1, 1)^\top$, alors v est solution de l'équation précédente morceau par morceau, c'est-à-dire :

$$\partial_{x_i} v = K_i {F_i}^{-1} v, \quad \forall i \in [\![1, n]\!]$$

et on obtient immédiatement le résultat. □

Remarque 2.15. La version avec « promoteurs rapides » correspondant à (2.9) s'obtient simplement en multipliant (2.24) par ρ, ce qui donne :

$$\underline{u}_\rho(x) = \frac{\exp(-\rho V(x))}{A_\rho \prod_{i=1}^{n} x_i(1 - x_i)}$$

où A_ρ est la constante de normalisation. On a alors $\underline{u}_\rho(x) \approx {A_\rho}^{-1} \exp(-\rho V(x))$ lorsque ρ est grand, d'où le fait qu'un tel V est souvent appelé *quasi-potentiel*. Intuitivement, la situation considérée dans la Proposition 2.14 est à rapprocher de la diffusion d'une particule dans le potentiel V, parfois appelée diffusion de Smoluchowski, définie par l'équation différentielle stochastique

$$dX_t = -\frac{1}{2} \nabla V(X_t) dt + dB_t$$

où $(B_t)_{t \geqslant 0}$ est un mouvement brownien standard de dimension n. En effet, on a dans les deux cas un système intégrable faisant apparaître directement V dans la dynamique, et la loi stationnaire de la diffusion est donnée par $u(x) \propto \exp(-V(x))$. Dans le cas du PDMP, c'est l'alternance entre les flots associés aux états ON et OFF qui joue le rôle de B_t. Il convient cependant de noter que contrairement à la diffusion, le processus $(E_t, X_t)_{t \geqslant 0}$ défini par (2.9) n'est pas réversible (Faggionato *et al.*, 2009).

Un des avantages de cette classe de modèles est que l'on peut comparer l'approximation de Hartree avec la loi exacte. En particulier, on peut faire le lien avec la pseudo-vraisemblance de Besag (1975), dont on rappelle la définition ci-dessous. Dans ce qui suit, étant donnés un vecteur aléatoire $X = (X_1, \ldots, X_n)$ de densité $p_\theta(x) = p_\theta(x_1, \ldots, x_n)$, formant un modèle statistique paramétré par θ, et deux parties $A, B \subset [\![1, n]\!]$ disjointes, on adopte la notation courante en statistique consistant à écrire $p_\theta(x_A | x_B) = p_\theta((x_i)_{i \in A} | (x_j)_{j \in B})$ la densité de $(X_i)_{i \in A}$ conditionnellement à $(X_j)_{j \in B}$.

Définition 2.16. La *pseudo-vraisemblance de Besag* d'un modèle statistique $p_\theta(x)$ est la fonction $\tilde{p}_\theta$ définie par

$$\tilde{p}_\theta(x) = \prod_{i=1}^{n} p_\theta(x_i | x_{-i}),$$

c'est-à-dire le produit des densités conditionnelles des X_i sachant $X_{-i} = (X_j)_{j \in [\![1,n]\!] \setminus \{i\}}$.

Intuitivement, on a de bonnes chances d'avoir $\tilde{p}_\theta(x) \approx p_\theta(x)$ lorsque les dépendances entre les variables X_i ne sont pas trop fortes, sachant que l'égalité est clairement vérifiée dans le cas indépendant, c'est-à-dire lorsque $p_\theta(x_i|x_{-i}) = p_\theta(x_i)$. Or on a typiquement une forme bien plus simple pour les densités conditionnelles $p_\theta(x_i|x_{-i})$ que pour la densité jointe $p_\theta(x)$: disposant d'une observation x, on peut alors gagner énormément de temps en maximisant $\theta \mapsto \tilde{p}_\theta(x)$ plutôt que $\theta \mapsto p_\theta(x)$. L'utilisation de cette pseudo-vraisemblance est aujourd'hui largement répandue, du fait des grands services qu'elle peut rendre dans certaines situations où la complexité de $p_\theta(x)$ serait rédhibitoire (Murphy, 2012). Cela ressemble en fait beaucoup à notre approximation de Hartree, et il est intéressant de constater que dans les cas où l'on sait calculer la loi stationnaire du réseau, les deux approches coïncident.

Corollaire 2.17. *Sous les hypothèses de la Proposition 2.14, l'approximation de Hartree $\underline{h}$ de la Définition 2.8 est égale à la pseudo-vraisemblance de Besag associée à $\underline{u}$.*

Démonstration. Il suffit de vérifier que $\underline{h}$ correspond au produit des conditionnelles de $\underline{u}$, ce qui est en fait une conséquence directe de (2.12) et de la Proposition 2.14. □

2.3.2 Modèles graphiques probabilistes

En choisissant V de la forme $V(x) = f_1(x_1) + \cdots + f_n(x_n)$ dans la Proposition 2.14, on couvre complètement le cas des gènes isolés avec rétroaction quelconque. Bien que ce cas soit déjà traité par Boxma *et al.* (2005), le point de vue vectoriel hybride fournit une preuve bien plus simple, en exploitant de manière fondamentale le fait que les matrices $K^{(i)}$ de l'équation maîtresse (2.4) admettent un vecteur propre constant. Le cas le plus intéressant est bien sûr celui où les gènes peuvent interagir : sur ce point, les réseaux de régulation concernés par la Proposition 2.14 sont rarement réalistes puisque la relation (2.24) impose une condition de symétrie sur les interactions, liée au fait que $\partial_{x_i}\partial_{x_j}V = \partial_{x_j}\partial_{x_i}V$ dès que V est deux fois différentiable. Néanmoins, il est concevable que certaines sous-classes particulières de ces réseaux soient numériquement assez proches de modèles bien plus réalistes, avec l'intérêt de pouvoir être étudiés en détail. Nous allons nous intéresser en particulier à des propriétés d'indépendance conditionnelle.

Soit $\mathcal{G}$ un graphe non orienté à n sommets notés $1, \ldots, n$. On écrit $i \sim j$ lorsque i et j sont voisins dans $\mathcal{G}$, c'est-à-dire quand $\mathcal{G}$ possède une arrête reliant i et j. Considérons les densités de la forme

$$p(x) \propto \prod_{1 \leqslant i \leqslant n} \phi_i(x_i) \prod_{\substack{1 \leqslant i < j \leqslant n \\ i \sim j}} \phi_{ij}(x_i, x_j) \tag{2.26}$$

où ϕ_i et ϕ_{ij} sont des fonctions à valeurs strictement positives. Il est facile de vérifier que si un vecteur aléatoire X admet $p(x)$ pour densité, alors pour tout $i \in [\![1, n]\!]$, X_i est indépendant des autres X_j conditionnellement à ses voisins. On dira alors que X (ou sa loi p) est un *champ de Markov* pour le graphe $\mathcal{G}$. L'intérêt de ce formalisme est d'être intuitif : si par exemple on considère une voie de signalisation $1 \to 2 \to 3$, on peut s'attendre à ce que la dépendance statistique du gène 3 par rapport au gène 1 provienne exclusivement du passage par le gène 2, ou en d'autres termes, que X_3 soit indépendant de X_1 sachant X_2. Noter d'une part que l'approximation de Hartree $\underline{h}$ vérifie précisément cette propriété par construction, et d'autre part qu'il est tout à fait possible que la loi stationnaire exacte ne la vérifie pas. Là encore, le cas résoluble apporte une réponse intéressante.

Corollaire 2.18. *Sous les hypothèses de la Proposition 2.14, si V peut s'écrire sous la forme*

$$V(x) = \sum_{1 \leqslant i \leqslant n} f_i(x_i) \; + \sum_{\substack{1 \leqslant i < j \leqslant n \\ i \sim j}} f_{ij}(x_i, x_j),$$

alors la loi des protéines (2.25) *est un champ de Markov pour le graphe $\mathcal{G}$.*

La démonstration est immédiate puisque l'on obtient $\underline{u}$ sous la forme (2.26) en passant à l'exponentielle. En faisant le lien avec l'Exemple 1.3, on a une explication assez intuitive : la forme de somme dans V correspond par (2.24) à des sommes dans a_i ou b_i, qui elles-mêmes traduisent des réactions se produisant *en parallèle* donc indépendamment les unes des autres, ce qui aboutit assez naturellement à des indépendances dans la loi stationnaire. L'exemple suivant correspond au cas où ces réactions sont de la forme $\mathsf{G}_i + \mathsf{P}_i + \mathsf{P}_j \to \mathsf{G}_i^* + \mathsf{P}_i + \mathsf{P}_j$.

Exemple 2.19 (Champ de Markov quadratique)**.** Si les fonctions d'interaction sont définies par

$$\frac{a_i(x)}{d_{1,i}} = \alpha_{i0} + \alpha_{ii} x_i + \sum_{\substack{1 \leqslant j \leqslant n \\ j \neq i}} \alpha_{ij} x_i x_j \quad \text{et} \quad \frac{b_i(x)}{d_{1,i}} = \beta_{i0}$$

avec $\alpha_{i0}, \beta_{i0} > 0$, $\alpha_{ii}, \alpha_{ij} \geqslant 0$ et $\alpha_{ij} = \alpha_{ji}$, alors on obtient

$$-V(x) = \sum_{i=1}^{n} \alpha_{ii} x_i \; + \sum_{1 \leqslant i < j \leqslant n} \alpha_{ij} x_i x_j \; + \sum_{i=1}^{n} \alpha_{i0} \ln(x_i) + \beta_{i0} \ln(1 - x_i).$$

Ainsi, le vecteur X des protéines en régime stationnaire est un champ de Markov pour le graphe $\mathcal{G}$ dont les arrêtes correspondent aux α_{ij} non nuls :

$$\underline{u}(x) = p_\alpha(x) \propto \exp\left(\sum_{i=1}^{n} \alpha_{ii} x_i \; + \sum_{1 \leqslant i < j \leqslant n} \alpha_{ij} x_i x_j \right) \prod_{i=1}^{n} {x_i}^{\alpha_{i0} - 1} (1 - x_i)^{\beta_{i0} - 1}.$$

Dans cet exemple de réseau, la loi stationnaire $\underline{u}$ est en fait un cas typique d'*auto-modèle* de Besag (1974) dans le sens où les densités conditionnelles ont toutes la même forme,[7] ici $p_\alpha(x_i | x_{-i}) \propto {x_i}^{\alpha_{i0} - 1} (1 - x_i)^{\beta_{i0} - 1} \exp(x_i [\alpha_{ii} + \sum_{j \neq i} \alpha_{ij} x_j])$. Cette propriété est intimement liée au fait que les densités p_α pour $\alpha_{ii}, \alpha_{ij} \geqslant 0$ forment une famille exponentielle, ce qui permet de construire des algorithmes d'inférence – exacts ou approchés – très efficaces, basés sur des résultats de convexité (Wainwright et Jordan, 2008). Du point de vue mécaniste, le principal défaut de l'Exemple 2.19 est que son comportement ne correspond absolument pas aux observations, pour les mêmes raisons que l'Exemple 1.3. L'objectif final de cette section est de retrouver un lien avec la forme issue du modèle de chromatine, en commençant par s'intéresser au cas de deux gènes en interaction symétrique.

Exemple 2.20. On se place dans le cas $n = 2$ et on considère la forme (1.11) pour $k_{\mathrm{on},i}$, avec $k_{\mathrm{off},i}$ constante. On suppose que pour $i, j \in \{1, 2\}$ avec $i \neq j$, on a les relations

$$m_{ii} = m_{ji} = \frac{k_{1,i} - k_{0,i}}{d_{1,i}}, \quad s_{ii} = \sigma_i \quad \text{et} \quad s_{ji} = \sigma_i \exp\left(-\frac{\theta_{ii}}{m_{ii}} \right)$$

7. On pourrait aussi parler de modèle auto-régressif spatial (donc sans orientation particulière), à ne pas confondre avec les modèles auto-régressifs temporels pour lesquels cette appellation est plus répandue.

où les σ_i sont les facteurs d'échelle des protéines intervenant dans (1.15). On aboutit alors à

$$a_i(x) = \frac{k_{0,i} + k_{1,i}\phi_{ij}(x_j)e^{\theta_{ii}}{x_i}^{m_{ii}}}{1 + \phi_{ij}(x_j)e^{\theta_{ii}}{x_i}^{m_{ii}}} \quad \text{et} \quad b_i(x) = k_{\text{off},i}$$

où $\phi_{ij}(x_j) = (1 + e^{\theta_{jj}+\theta_{ij}}{x_j}^{m_{jj}})/(1 + e^{\theta_{jj}}{x_j}^{m_{jj}})$. Enfin, on suppose que $\theta_{12} = \theta_{21}$. Alors en notant $a_{0,i} = k_{0,i}/d_{1,i}$, $a_{1,i} = m_{ii}$, $a_{2,i} = k_{\text{off},i}/d_{1,i}$ et $\theta_i = \theta_{ii}$, on obtient :

$$-V(x) = \ln\left(\sum_{0\leqslant z_1,z_2\leqslant 1} e^{\theta_{12}z_1z_2}\prod_{i=1}^{2} e^{\theta_i z_i}{x_i}^{a_{0,i}+a_{1,i}z_i}\right) + \sum_{i=1}^{2} a_{2,i}\ln(1-x_i)$$

et finalement

$$\underline{u}(x) = p_\theta(x) \propto \sum_{0\leqslant z_1,z_2\leqslant 1} e^{\theta_1 z_1+\theta_2 z_2+\theta_{12}z_1z_2}\prod_{i=1}^{2}{x_i}^{a_{0,i}+a_{1,i}z_i-1}(1-x_i)^{a_{2,i}-1}.$$

En d'autres termes, on peut écrire $\underline{u}$ sous la forme

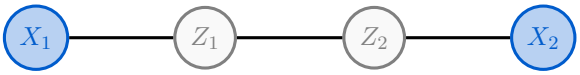
$$\underline{u} \propto f_{00} + e^{\theta_1}f_{10} + e^{\theta_2}f_{01} + e^{\theta_1+\theta_2+\theta_{12}}f_{11}$$

où les fonctions $f_{z_1z_2}$ sont définies par $f_{z_1z_2}(x_1,x_2) = \prod_{i=1}^{2}{x_i}^{a_{0,i}+a_{1,i}z_i-1}(1-x_i)^{a_{2,i}-1}$.

On constate que la loi $\underline{u}$ correspondant à cet exemple n'est pas un champ de Markov à cause de la somme sur z_1 et z_2. En revanche, celle-ci apparaît clairement comme un mélange de lois plus simples : cela suggère de « démarginaliser », i.e. interpréter $\underline{u}$ comme la densité marginale de la composante X d'un vecteur (X,Z) dont la loi est donnée par

$$p_\theta(x,z) \propto \exp\left(\sum_{i=1}^{2}\theta_i z_i + \theta_{12}z_1z_2\right)\prod_{i=1}^{2}{x_i}^{a_{0,i}+a_{1,i}z_i-1}(1-x_i)^{a_{2,i}-1}.$$

Il est maintenant clair que les $p_\theta(x,z)$ forment une famille exponentielle en θ. De plus, le vecteur aléatoire (X,Z) peut s'interpréter comme un champ de Markov dont le graphe est



et qui est caractérisé par les lois conditionnelles suivantes :

$$\mathcal{L}(X_1|Z_1) = \text{Beta}(a_{0,1}+a_{1,1}Z_1, a_{2,1}), \quad \mathcal{L}(Z_1|X_1,Z_2) = \mathcal{B}\left(\frac{\exp(\theta_1+\theta_{12}Z_2){X_1}^{a_{1,1}}}{1+\exp(\theta_1+\theta_{12}Z_2){X_1}^{a_{1,1}}}\right),$$

$$\mathcal{L}(X_2|Z_2) = \text{Beta}(a_{0,2}+a_{1,2}Z_2, a_{2,2}), \quad \mathcal{L}(Z_2|X_2,Z_1) = \mathcal{B}\left(\frac{\exp(\theta_2+\theta_{12}Z_1){X_2}^{a_{1,2}}}{1+\exp(\theta_2+\theta_{12}Z_1){X_2}^{a_{1,2}}}\right),$$

où $\mathcal{B}(p)$ désigne la loi de Bernoulli de paramètre p. On peut ainsi voir ce champ de Markov comme un « auto-modèle Beta-logistique » qui s'avère bien mieux décrire les données que le champ de Markov quadratique de l'Exemple 2.19 : l'idée sera de l'étendre à un nombre quelconque de gènes. Avant cela, profitons-en pour préciser une situation plus générale où il est intéressant de faire apparaître la variable Z.

Proposition 2.21. *On suppose que les fonctions b_i sont constantes et qu'il existe une distribution μ sur un ensemble fini $\mathcal{S} \subset (\mathbb{R}_+^*)^n$ telle que les fonctions a_i peuvent s'écrire sous la forme*

$$a_i(x) = d_{1,i} \frac{\sum_{z\in\mathcal{S}} z_i x^z \mu(z)}{\sum_{z\in\mathcal{S}} x^z \mu(z)}$$

avec $x^z = {x_1}^{z_1} \cdots {x_n}^{z_n}$. Alors la densité stationnaire des protéines est donnée par

$$\underline{u}(x) = p(x) \propto \sum_{z\in\mathcal{S}} \mu(z) \prod_{i=1}^{n} {x_i}^{z_i-1}(1-x_i)^{b_i/d_{1,i}-1}.$$

De plus, $a_i(x)$ est alors l'espérance de $d_{1,i}z_i$ sous la loi conditionnelle $p(z|x) \propto x^z\mu(z)$.

Démonstration. On applique simplement la Proposition 2.14 avec

$$-V(x) = \sum_{z\in\mathcal{S}} x^z\mu(z) + \sum_{i=1}^{n} \frac{b_i}{d_{1,i}} \ln(1-x_i).$$

On peut ensuite introduire une variable Z de façon à définir la loi jointe de (X, Z) par $p(x, z) \propto \mu(z) \prod_{i=1}^n {x_i}^{z_i-1}(1-x_i)^{b_i/d_{1,i}-1}$ puis remarquer que puisque $p(z|x) \propto x^z\mu(z)$, on obtient a fortiori $a_i(x) = \mathbb{E}[Z_i|X = x]$. □

Cette situation semble être le meilleur cadre d'application de la Proposition 2.14, dans la perspective d'un modèle de réseau le plus réaliste possible avec une loi stationnaire explicite. Plusieurs indices suggèrent en effet que la régulation des gènes porte surtout sur la fréquence des bursts (Viñuelas *et al.*, 2013 ; Senecal *et al.*, 2014 ; Fukaya *et al.*, 2016), et il est donc assez naturel de considérer b_i constant et a_i sous la forme d'une fonction bornée de type Hill comme dans le cas de notre modèle de chromatine (1.11). La Proposition 2.21 décrit justement ce type d'interactions, et l'interprétation de $a_i(x)$ comme une espérance est tout à fait cohérent avec l'approche physique de la section 1.2.

2.3.3 Auto-modèle Gamma-Binomial

On adapte maintenant l'Exemple 2.20 à un nombre quelconque de gènes, en remarquant que la loi conditionnelle de Z_i peut se généraliser à une loi binomiale en se basant sur l'Exemple 2.6. Plus précisément, on applique la Proposition 2.21 à la distribution

$$\mu(z) = \exp\left(\sum_{i=1}^{n} \theta_i z_i + \sum_{1\leqslant i<j\leqslant n} \theta_{ij} z_i z_j\right) \prod_{i=1}^{n} \binom{c_i}{z_i}$$

pour $c_i \in \mathbb{N}^*$ et $z \in \mathcal{S} = \prod_{i=1}^n [\![0, c_i]\!]$, au changement de variable $z_i \mapsto a_{0,i} + a_{1,i}z_i$ près et avec $b_i = d_{1,i}a_{2,i}$ où $a_{0,i}$, $a_{1,i}$ et $a_{2,i}$ sont définis comme dans l'Exemple 2.20. Noter qu'il existe bel et bien un modèle de chromatine alternatif (encore réversible) qui donnerait précisément cette forme. Par ailleurs, on se place dans le régime bursty en remplaçant les lois Beta par des lois Gamma sans changer l'idée du modèle, c'est-à-dire en faisant l'approximation

$$(1-x_i)^{a_{2,i}-1} = \exp((a_{2,i}-1)\ln(1-x_i)) \approx \exp(-a_{2,i}x_i)$$

qui est valable lorsque $a_{2,i} \gg 1$ et $a_{2,i} \gg a_{0,i} + a_{1,i}c_i$ (la masse se concentrant alors vers $x_i \ll 1$). Finalement, on aboutit au modèle statistique suivant.

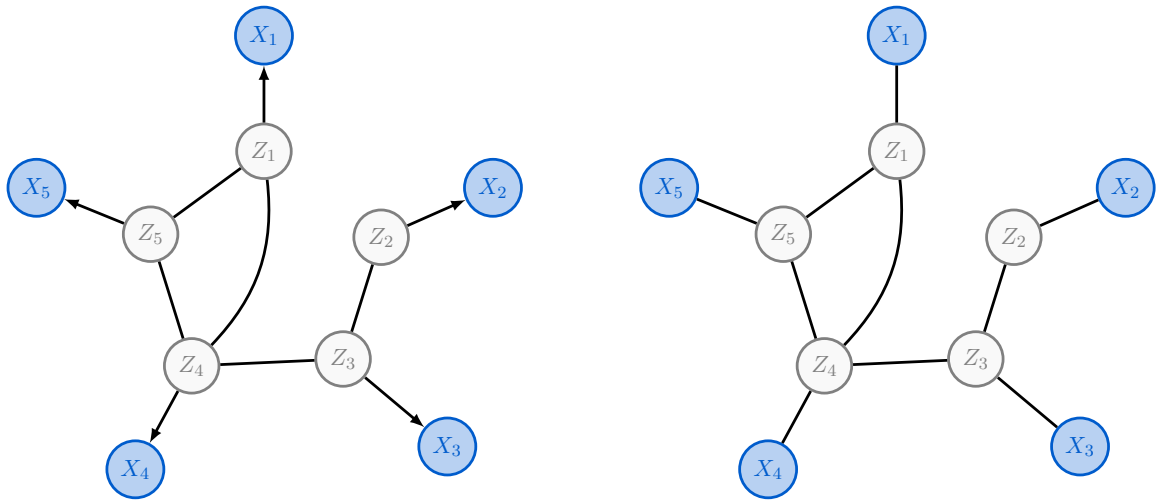


Figure 2.3 – Exemple de graphe associé à l'auto-modèle Gamma-Binomial dans le cas $n = 5$. Ce modèle peut être interprété soit comme un champ de Markov caché $Z \to X$ en déterminant d'abord la loi de Z (gauche), soit comme un champ de Markov (X, Z) partiellement observé (droite).

Définition 2.22. Soit $\theta = (\theta_{ij})_{1 \leqslant i,j \leqslant n} \in \mathcal{S}_n(\mathbb{R})$ une matrice symétrique. L'*auto-modèle Gamma-Binomial* est défini par la densité

$$p_\theta(x, z) \propto \exp\left(\sum_{i=1}^n \theta_i z_i + \sum_{1 \leqslant i < j \leqslant n} \theta_{ij} z_i z_j\right) \prod_{i=1}^n \binom{c_i}{z_i} {x_i}^{a_{0,i} + a_{1,i} z_i - 1} e^{-a_{2,i} x_i} \tag{2.27}$$

où $a_{0,i}$, $a_{1,i}$, $a_{2,i} \in \mathbb{R}_+^*$ et $c_i \in \mathbb{N}^*$ sont vus comme des hyperparamètres fixés.

En d'autres termes, on a les lois conditionnelles

$$\mathcal{L}(X|Z = z) = \bigotimes_{i=1}^n \text{Gamma}\left(a_{0,i} + a_{1,i} z_i, a_{2,i}\right),$$

$$\mathcal{L}(Z_i|X_i = x_i, Z_{-i} = z_{-i}) = \mathcal{B}\left(c_i, \phi\left(e^{\theta_i + \sum_{j \neq i} \theta_{ij} z_j} {x_i}^{a_{1,i}}\right)\right)$$

où $\phi(s) = s/(1+s)$ et $\mathcal{B}(c, p)$ désigne la loi binomiale de paramètres $c \in \mathbb{N}^*$ et $p \in [0, 1]$. La loi jointe $\mathcal{L}(X, Z)$ peut alors être interprétée de deux façons différentes :

- comme un champ de Markov caché, car les X_i sont indépendants sachant Z et il est facile de vérifier que Z est un champ de Markov pour le graphe associé à θ ;
- comme un champ de Markov partiellement observé, en considérant le graphe non orienté associé à la Définition 2.22 sans introduire de hiérarchie de Z vers X.

Noter que la deuxième option est plus pratique dans le sens où elle fait apparaître les lois Gamma et Binomiale qui sont bien connues, tandis que la loi marginale de Z a une forme moins simple. La Figure 2.3 montre les deux représentations graphiques correspondantes pour $n = 5$, dans le cas où seuls θ_{14}, θ_{15}, θ_{23}, θ_{34} et θ_{45} sont non nuls.

Remarque 2.23. Si l'on voulait appliquer à la lettre la simplification considérée dans la section 2.1, il faudrait utiliser à la fois X et Z comme des variables cachées et considérer que les observations Y sont distribuées selon la loi conditionnelle (2.2). En fait, l'introduction de Z a mis en valeur l'importance des modes de fréquence des promoteurs : comme par

construction l'ARNm suit le même mode de fréquence que les protéines, il est assez naturel de se dire que la variable X de l'auto-modèle Gamma-Binomial peut également décrire directement l'ARNm, quitte à changer les hyperparamètres. On vérifiera en pratique que cela fonctionne très bien numériquement.[8] Noter que cette méthode reste bien moins grossière que le fait de considérer l'ARNm comme « proxy » pour les protéines, puisque l'on passe par l'introduction d'une variable cachée Z décrivant les modes de fréquence des promoteurs. En outre, on pourra remarquer que plusieurs approches empiriques ont abouti à des modèles de mélange très proches (Gu *et al.*, 2015 ; Ghazanfar *et al.*, 2016 ; Bartlett *et al.*, 2017), sans toutefois faire apparaître des interactions sous la forme d'un champ de Markov.

Pour conclure ce chapitre, nous allons montrer que l'auto-modèle Gamma-Binomial est *identifiable* en θ dans le sens classique où l'application $\theta \mapsto \mathcal{L}(X|\theta)$ est injective (Z étant cachée), ce qui demeure la principale justification des approximations effectuées jusqu'ici. Comme indiqué dans la Définition 2.22, on considère que $a_{0,i}$, $a_{1,i}$, $a_{2,i}$ et c_i sont connus et fixés. On s'intéresse dans un premier temps à des modèles de Markov cachés Z généraux avec lois d'émission Gamma, i.e. tels que $\mathcal{L}(X|Z)$ est un produit de lois Gamma.

Lemme 2.24. *Soient $a_i, b_i, c_i > 0$ fixés pour $i \in [\![1, n]\!]$. Soient $Z_1 = (Z_{1,1}, \dots, Z_{1,n})$ et $Z_2 = (Z_{2,1}, \dots, Z_{2,n})$ des vecteurs aléatoires à valeurs dans $\mathbb{N}^n$, et soient $X_1 = (X_{1,1}, \dots, X_{1,n})$ et $X_2 = (X_{2,1}, \dots, X_{2,n})$ des vecteurs aléatoires dont les lois sont données par*

$$\mathcal{L}(X_1|Z_1) = \bigotimes_{i=1}^{n} \mathrm{Gamma}(a_i Z_{1,i} + b_i, c_i) \quad et \quad \mathcal{L}(X_2|Z_2) = \bigotimes_{i=1}^{n} \mathrm{Gamma}(a_i Z_{2,i} + b_i, c_i).$$

Alors $\mathcal{L}(X_1) = \mathcal{L}(X_2)$ si et seulement si $\mathcal{L}(Z_1) = \mathcal{L}(Z_2)$.

Démonstration. On a par définition $p(x|z) = \prod_{i=1}^{n} {c_i}^{a_i z_i + b_i} \Gamma(a_i z_i + b_i)^{-1} {x_i}^{a_i z_i + b_i - 1} e^{-c_i x_i}$ où x (resp. z) désigne la valeur prise par $X = X_1$ ou X_2 (resp. $Z = Z_1$ ou Z_2). On utilise alors la transformée de Laplace ϕ_X, en remarquant que

$$\phi_X(s) = g(s)\phi_Z(h(s)), \quad \forall s \in \mathbb{R}_+^n$$

où $g(s) = \prod_{i=1}^{n} (1 + s_i/c_i)^{-c_i}$ et $h(s) = (a_1 \ln(1 + s_1/c_1), \dots, a_n \ln(1 + s_n/c_n))$. La conclusion vient du fait que $g > 0$ et ne dépend pas de Z et que h est inversible sur $\mathbb{R}_+^n$. □

Proposition 2.25. *La loi $p_\theta(x)$ de l'auto-modèle Gamma-Binomial est identifiable en θ.*

Démonstration. Considérons la loi $p_\theta(z)$ de la variable cachée. En intégrant (2.27) en x, on obtient

$$p_\theta(z) \propto \exp\left(\sum_{i=1}^{n} \theta_i z_i + \sum_{1 \leqslant i < j \leqslant n} \theta_{ij} z_i z_j \right) \prod_{i=1}^{n} \binom{c_i}{z_i} \frac{\Gamma(a_{0,i} + a_{1,i} z_i)}{{a_{2,i}}^{a_{0,i} + a_{1,i} z_i}}$$

de sorte que les $p_\theta(z)$ forment clairement une famille exponentielle minimale en θ. La loi de Z est donc identifiable en θ, et l'on peut conclure grâce au Lemme 2.24. □

8. On n'en est plus à une approximation près, et la référence reste le modèle complet défini au chapitre 1.

Chapitre 3

Généralisation du modèle à deux états et aspects algébriques

Dans ce chapitre, nous laissons l'inférence de coté et revenons simultanément sur deux aspects évoqués lors de l'introduction du modèle à deux états :

- Le "nombre d'états" du promoteur. Le principe d'un unique état actif avec un temps d'attente distribué selon une loi exponentielle (de paramètre k_{off}) semble faire consensus chez les biologistes qui étudient la transcription (Larson, 2011). En revanche, le temps d'attente dans l'état inactif ne semble pas toujours suivre une loi exponentielle mais plutôt une loi Gamma pour certains gènes, suggérant pour rester markovien la présence d'états inactifs intermédiaires. Il est possible d'étudier ces promoteurs "réfractaires" en généralisant naturellement le modèle à deux états en un *modèle à n états*.
- Le lien entre le point de vue du processus markovien de sauts et celui du PDMP. On pouvait déjà se douter, au vu des lois stationnaires présentées à la section 1.1.4 du premier chapitre (Beta-Poisson d'un coté et Beta de l'autre), qu'il n'y a pas seulement un lien "approximatif" entre ces deux processus, mais également un lien algébrique sans passage à la limite. Nous verrons que ce lien peut être assez agréablement interprété en termes de "changement de base" dans l'espace vectoriel des mesures boréliennes finies sur $\mathbb{R}_+$ et de sous-espace stable pour le semi-groupe associé à l'équation maîtresse du processus de sauts. Le principe est assez simple mais ne semble pas si connu, et surtout il motive vraiment l'utilisation de ce genre de stratégie dans un contexte plus large.

Ce chapitre est basé sur un article soumis : nous considérons un modèle généralisé dans lequel la transcription dépend d'un promoteur avec un nombre quelconque d'états – incluant le modèle à deux états et les promoteurs réfractaires comme cas particuliers – et nous nous intéressons à l'obtention de la loi stationnaire exacte. En partant de plusieurs approches précédemment développées et en les unifiant, nous parvenons à simplifier et généraliser les résultats existants. En particulier, le processus de sauts original s'avère profondément relié à un PDMP multivarié qui pourrait aussi avoir une interprétation dans d'autres domaines que la biologie. Pour une configuration très particulière du promoteur, nous montrons que la loi stationnaire de ce PDMP est la loi de Dirichlet. Dans le cas général, il s'avère que les marginales étendent la classe bien connue des produits de variables indépendantes de loi Beta, en faisant intervenir des paramètres complexes directement reliés aux propriétés spectrales de la matrice de transition du promoteur. Finalement, nous illustrons ces résultats par des exemples biologiquement plausibles.

Introduction

Gene expression within a cell, that is, transcription of specific regions of its DNA into mRNA molecules (to be then translated into proteins), is now well-acknowledged to be a stochastic phenomenon resulting from a set of various chemical reactions, some of which involving species that are only present in very small quantities. As a relevant compromise, a gene is usually described by its *promoter* and gene expression models consist of two reactions occurring in parallel: creation of mRNA by the promoter and degradation of mRNA (Dattani et Barahona, 2017). When both creation and degradation have constant rates, one gets a standard birth-death process that has a Poisson stationary distribution. Whereas such an elementary degradation is often satisfactory, the creation part is somewhat of a hot topic and more sophisticated models have been proposed, depending on the biological context (Coulon *et al.*, 2010; Zoller *et al.*, 2015; Dattani et Barahona, 2017).

For instance, measures of gene expression in individual, isogenic cells in the same environment typically show a heavy-tailed distribution with a clearly non-Poisson variance (Albayrak *et al.*, 2016; Richard *et al.*, 2016). The simplest model to account for this fact is the well-established "two-state model", which is a birth-death process in random environment (Peccoud et Ycart, 1995; Kim et Marioni, 2013). As suggested by the name, such a promoter has one *active* state in which the mRNA creation rate is positive and one *inactive* state in which the creation rate is zero. Depending on the switching rates between states, the time spent in the active one can be short enough to generate so-called "bursty" mRNA dynamics (Herbach *et al.*, 2017), leading to a much more realistic distribution than the one-state previous model.

The two-state model has the great advantage of being tractable, and sometimes it can even be physically justified as a relevant first-order approximation (Chong *et al.*, 2014). However, as single-cell experiments become more precise, it appears that some promoters cannot be described by only two states because their inactive period has a non-exponential distribution with a positive mode (Zoller *et al.*, 2015). Such cases suggest a "refractory" behaviour, meaning that after each active phase, the promoter has to progress through several inactive states before getting active again.

These observations have motivated the introduction of "multistate" promoters, each state being associated with a particular rate of mRNA creation (Coulon *et al.*, 2010; Zhou *et al.*, 2012; Innocentini *et al.*, 2013; Dattani et Barahona, 2017). Accordingly, we consider a promoter with n states ($n \geqslant 2$) represented by chemical species $\mathsf{S}_1, \dots, \mathsf{S}_n$, with transitions between states such that molecule numbers always satisfy $[\mathsf{S}_i] \in \{0, 1\}$ for all i and $[\mathsf{S}_1] + \cdots + [\mathsf{S}_n] = 1$. Then, representing mRNA by a species M, the expression model is defined by the following system of elementary reactions:

$$\begin{cases} \mathsf{S}_i \xrightarrow{r_{i,j}} \mathsf{S}_j & \text{for } i, j \in [\![1, n]\!],\ i \neq j \\ \mathsf{S}_i \xrightarrow{u_i} \mathsf{S}_i + \mathsf{M} & \text{for } i \in [\![1, n]\!] \\ \mathsf{M} \xrightarrow{d_0} \varnothing \end{cases} \tag{3.1}$$

with rates $r_{i,j} \geqslant 0$, $u_i \geqslant 0$ and $d_0 > 0$. Importantly, two distinct scenarios can be considered for this model:

1. the general case (e.g., Coulon *et al.*, 2010; Innocentini *et al.*, 2013);
2. the particular case where only one u_i is nonzero (e.g., Zhou *et al.*, 2012; Zoller *et al.*,

2015; Dattani et Barahona, 2017).

Promoters that belong to the second case can be interpreted as having exactly one active state and $n-1$ inactive states: in line with the intuition presented above, we shall call them *refractory* in the present work. Note that in this view, the two-state model corresponds to a "trivial" refractory promoter.

Our main interest here is the stationary distribution of the mRNA quantity $[\mathsf{M}]$. In (Innocentini *et al.*, 2013), the authors provide a general but implicit formula based on a recurrence relation, focusing on multimodality induced by distinct u_i values. On the other hand, the authors in (Zhou *et al.*, 2012) consider some particular refractory promoters (transition graph forming a cycle) and express the distribution in terms of generalized hypergeometric functions (Slater, 1966), providing an implicit way to derive the parameter values. A further step is achieved in (Dattani, 2016), where parameters are explicitly derived in a more particular case (irreversible cycle).

In this work, we propose to gather, simplify and extend these results by adopting a unified viewpoint: the underlying philosophy is to "break down the noise", that is, to decompose the complexity of the distribution into simpler layers. As suggested in (Dattani et Barahona, 2017), we use the Poisson representation (Gardiner et Chaturvedi, 1977) of system (3.1), which allows for combining approaches in (Innocentini *et al.*, 2013) and (Zhou *et al.*, 2012) by introducing a piecewise-deterministic Markov process (PDMP). First, we reinterpret the main result of (Innocentini *et al.*, 2013) as a projection of the PDMP joint distribution. We show a simplistic situation where this distribution is Dirichlet, yet providing some interesting insight into the general case. Second, we simplify the main result of (Zhou *et al.*, 2012) concerning cyclic refractory promoters, and generalize it to any refractory promoter by only assuming irreducible dynamics (i.e., for any $i \neq j$, there exists a path of reactions from S_i to S_j with positive reaction rates). This refractory case exactly corresponds to marginals of the previous joint distribution. Interestingly, the resulting class of univariate distributions generalizes the one consisting of products of Beta-distributed random variables, which also arises in statistics (Dufresne, 2010) and mathematical physics (Dunkl, 2013). It is characterized by a set of parameters that are potentially complex and directly relate to spectral properties of the promoter transition matrix.

This chapter is organized as follows. The mathematical formulation of system (3.1) is introduced in section 3.1 and its Poisson representation is detailed in section 3.2. Then, the underlying multivariate PDMP is presented in section 3.3 and the complete solution for refractory promoters is given in section 3.4. Finally, applications and a discussion follow, as well as two short appendices 3.A and 3.B with technical details.

3.1 Basic mathematical model

For $t \geqslant 0$, let E_t and M_t respectively denote the promoter state ($E_t = i$ if $[\mathsf{S}_i] = 1$, $i \in [\![1, n]\!]$) and the mRNA level ($M_t = [\mathsf{M}]$) at time t. Throughout this work, we adopt a semi-vectorial notation by encoding promoter states as components of $\mathbb{R}^n$ while keeping mRNA as a scalar: this will make our computations much easier and will essentially reduce the results to linear algebra. We assume that system (3.1) follows standard *stochastic mass-action* kinetics, that is, $(E_t, M_t)_{t \geqslant 0}$ is a jump Markov process with state space $[\![1, n]\!] \times \mathbb{N}$ and

generator $\mathcal{L}$ defined by

$$\mathcal{L}f(k) = d_0 k[f(k-1) - f(k)] + C[f(k+1) - f(k)] + Qf(k) \tag{3.2}$$

where $f(k) = (f_1(k), \dots, f_n(k))^\top \in \mathbb{R}^n$ represents functions $f : [\![1, n]\!] \times \mathbb{N} \to \mathbb{R}$, $C = \mathrm{Diag}(u_1, \dots, u_n) \in \mathbb{R}^{n \times n}$ contains creation rates and $Q \in \mathbb{R}^{n \times n}$ is the *promoter transition matrix* given by

$$Q_{i,j} = r_{i,j} \quad \text{for } i \neq j \quad \text{and} \quad Q_{i,i} = -\sum_{j \neq i} r_{i,j}.$$

In practice, we shall focus on distributions (meaning probability measures here) and therefore consider the adjoint operator of $\mathcal{L}$, denoted by Ω and defined by

$$\Omega g(k) = d_0[(k+1)g(k+1) - kg(k)] + C[g(k-1)\mathbb{1}_{k>0} - g(k)] + Hg(k) \tag{3.3}$$

where $H = Q^\top$ and $g = (g_1, \dots, g_n)^\top$ now stands for distributions g on $[\![1, n]\!] \times \mathbb{N}$. The distribution $p(t) = \mathbb{P}_{(E_t, M_t)}$, represented by $p(t) = (p_1(\cdot, t), \dots, p_n(\cdot, t))^\top$ where $p_i(k, t) = \mathbb{P}(E_t = i, M_t = k)$, then evolves according to the well-known Kolmogorov forward equation:

$$\frac{dp}{dt} = \Omega p \tag{3.4}$$

which is often called *master equation* in this context. Note that (3.4) is the same master equation as in (Coulon *et al.*, 2010) and (Innocentini *et al.*, 2013). Also, it is a natural generalization of the master equation considered in (Zhou *et al.*, 2012), which corresponds here to *cyclic refractory promoters* (i.e., only one u_i is nonzero and the undirected graph induced by H is a n-cycle). See section 3.4 for a graphical representation of cyclic and general refractory promoters.

As mentioned above, we assume that Q is irreducible (and thus also H): this is sufficient to ensure that $p(t)$ converges as $t \to \infty$ to a unique stationary distribution (see Peccoud et Ycart, 1995 and references therein), which will be our main object of interest. Finally, we set $d_0 = 1$, say in h^{-1}, without loss of generality (equivalent to dividing (3.2) and (3.3) by d_0 and rescaling time by $1/d_0$) so the model is completely parametrized by

$$r = (r_{i,j})_{i,j \in [\![1,n]\!],\, i \neq j} \quad \text{and} \quad u = (u_1, \dots, u_n).$$

3.2 Poisson representation

In this section, we motivate the introduction of an underlying process that is not only useful for computations, but also arises naturally as a fundamental part of the original process $(E_t, M_t)_{t \geqslant 0}$. Our approach is based on the Poisson representation, initially introduced by Gardiner et Chaturvedi (1977) as a powerful ansatz-based technique for solving master equations. As emphasized by the authors, this representation is particularly adapted to chemical birth-death processes because of the particular jump rate form implied by stochastic mass-action kinetics. In our case, there is something more as the representation reveals an actual "hidden layer" that happens to be a piecewise-deterministic Markov process.

3.2.1 Notation and definitions

In all that follows, $\mathcal{M}(\mathbb{R}_+)$ and $\mathcal{M}(\mathbb{N})$ denote the real vector spaces of finite signed measures on $\mathbb{R}_+ = [0, \infty)$ and $\mathbb{N} = \{0, 1, 2, \ldots\}$. When they are nonnegative and have their total mass equal to 1, such measures are standard probability measures, termed *distributions* here. It is worth mentioning that an actual, precise consideration of spaces is not the point of this work, but the reader may find some details in Appendix 3.A. Note that $\mathcal{M}(\mathbb{N})$ simply corresponds to the space of real sequences whose series is absolutely convergent.

Intuitively, $\mathcal{M}(\mathbb{R}_+)$ and $\mathcal{M}(\mathbb{N})$ will describe mRNA levels. Let us now introduce three transforms that will be used extensively in this chapter. Given $\mu \in \mathcal{M}(\mathbb{R}_+)$, we define the *Laplace transform* L_μ by

$$L_\mu(s) = \int_0^\infty e^{sx} d\mu(x), \quad \forall s \in (-\infty, 0]. \tag{3.5}$$

Similarly, given $p \in \mathcal{M}(\mathbb{N})$, we consider the *generating function* G_p defined by

$$G_p(z) = \sum_{k=0}^{\infty} p(k) z^k, \quad \forall z \in [-1, 1]. \tag{3.6}$$

The last one is the starting point of the Poisson representation: for $\mu \in \mathcal{M}(\mathbb{R}_+)$, we define $P\mu \in \mathcal{M}(\mathbb{N})$ by

$$P\mu(k) = \int_0^\infty \frac{x^k}{k!} e^{-x} d\mu(x), \quad \forall k \in \mathbb{N}, \tag{3.7}$$

which gives us a linear operator $P : \mathcal{M}(\mathbb{R}_+) \to \mathcal{M}(\mathbb{N})$. We may term this operator the *Poisson transform* and call its image $\mathcal{P}$ the space of *Poisson mixtures*. When μ has a density f with respect to the Lebesgue measure on $\mathbb{R}_+$, we consistently set $Pf = P\mu$. The operator P clearly preserves total mass, that is, $P\mu(\mathbb{N}) = \mu(\mathbb{R}_+)$. Moreover, $\mu \geqslant 0$ implies $P\mu \geqslant 0$ and thus P maps distributions to distributions (such operators are sometimes called *stochastic*, see Rudnicki et Tyran-Kamińska, 2017). In this case, the most important fact is that one can draw $Y \sim P\mu$ using the hierarchical (aka bayesian) model:

$$\begin{aligned} X &\sim \mu \\ Y|X &\sim \text{Poisson}(X) \end{aligned}$$

where the second line stands for the conditional distribution $\mathbb{P}_{Y|X} = \text{Poisson}(X)$. In this sense, $P\mu$ is a *mixture of Poisson distributions* where μ is the mixing distribution.

Remark 3.1. It will be useful to extend the definition of $L_\mu(s)$ to $s \in \mathbb{C}$. Given $\mu \in \mathcal{M}(\mathbb{R}_+)$, by standard results of complex analysis L_μ is holomorphic on the half plane $\{s \in \mathbb{C} \mid \text{Re}(s) < 0\}$. If μ is compactly supported (e.g., $\mu = \mathbb{P}_X$ with $X \in [0, 1]$), then L_μ can be defined on $\mathbb{C}$ and is an entire function.

The basic Poisson representation consists in finding an evolution equation for μ by assuming the form $P\mu$ in equation (3.4), implicitly expecting μ to be simpler: this approach hence benefits from a remarkable probabilistic interpretation, in contrast to many other ansatz techniques commonly used to solve such equations. Besides, the following result noted by Feller (1943) enlightens the correspondence between the generating function and the Laplace transform in the context of Poisson mixtures. Importantly for us, it implies that P is injective.

Lemma 3.2. *Let $\mu \in \mathcal{M}(\mathbb{R}_+)$. Then for all $z \in [-1, 1]$,*

$$G_{P\mu}(z) = L_\mu(z - 1).$$

In particular, P induces a linear isomorphism from $\mathcal{M}(\mathbb{R}_+)$ onto $\mathcal{P}$.

Proof. By Fubini's theorem (valid for $z \in [-1, 1]$) applied on measures μ^+ and μ^- of the Jordan decomposition $\mu = \mu^+ - \mu^-$, we directly get

$$G_{P\mu}(z) = \sum_{k=0}^{\infty} \int_0^\infty \frac{(zx)^k}{k!} e^{-x} d\mu(x) = \int_0^\infty e^{(z-1)x} d\mu(x) = L_\mu(z - 1).$$

Hence, if $P\mu = 0$, then $L_\mu(z) = 0$ for all $z \in [-2, 0]$, and thus also for all $z \in (-\infty, 0]$ by Remark 3.1. As $\mu \mapsto L_\mu$ is injective on $\mathcal{M}(\mathbb{R}_+)$ (e.g., Klenke, 2014, Theorem 15.6), it follows that $\mu = 0$. Hence, P is injective and we have $\mathcal{M}(\mathbb{R}_+) \simeq \mathcal{P}$. □

Remark 3.3. Having this result in mind, it is no surprise that a very common way to solve master equations such as (3.4) consists in deriving an evolution equation for the generating function and then making the change of variable $s = z - 1$. It seems to be less known that the outcome is nothing but the evolution equation satisfied by the Laplace transform of the mixing distribution within the Poisson representation.

Finally, we just need to extend $\mathcal{M}(\mathbb{R}_+)$ and $\mathcal{M}(\mathbb{N})$ in order to add a description of promoter states. In line with the semi-vectorized definitions of $\mathcal{L}$ and Ω in (3.2) and (3.3), we represent finite signed measures on $[\![1, n]\!] \times \mathbb{R}_+$ and $[\![1, n]\!] \times \mathbb{N}$ by

$$\mathcal{M}_n(\mathbb{R}_+) = \mathcal{M}(\mathbb{R}_+)^n \quad \text{and} \quad \mathcal{M}_n(\mathbb{N}) = \mathcal{M}(\mathbb{N})^n.$$

The Laplace transform and the generating function are then naturally extended as

$$L_\mu = (L_{\mu_1}, \ldots, L_{\mu_n})^\top \quad \text{and} \quad G_p = (G_{p_1}, \ldots, G_{p_n})^\top$$

for $\mu \in \mathcal{M}_n(\mathbb{R}_+)$ and $p \in \mathcal{M}_n(\mathbb{N})$, as well as $P : \mathcal{M}_n(\mathbb{R}_+) \to \mathcal{M}_n(\mathbb{N})$ whose image is now denoted by $\mathcal{P}_n$. Clearly, Lemma 3.2 still holds and P induces an isomorphism from $\mathcal{M}_n(\mathbb{R}_+)$ onto $\mathcal{P}_n$.

3.2.2 Distribution viewpoint

The most intuitive strategy to derive the Poisson representation consists in directly injecting Poisson mixtures into equation (3.4) and then integrating by parts (see appendix 3.B.1 for details). If the boundary terms vanish, one finds that $p = P\mu$ is solution of (3.4) if and only if

$$\partial_t \mu = \partial_x(x\mu) - C\partial_x\mu + H\mu. \tag{3.8}$$

This is the key idea in (Gardiner et Chaturvedi, 1977): instead of using a series expansion, we obtain an *exact* representation as a time-dependent mixture of Poisson distributions. In our case, the evolution equation (3.8) satisfied by μ is a Kolmogorov forward equation associated with the new operator

$$\widetilde{\Omega}\mu = -\partial_x[(C - xI)\mu] + H\mu, \tag{3.9}$$

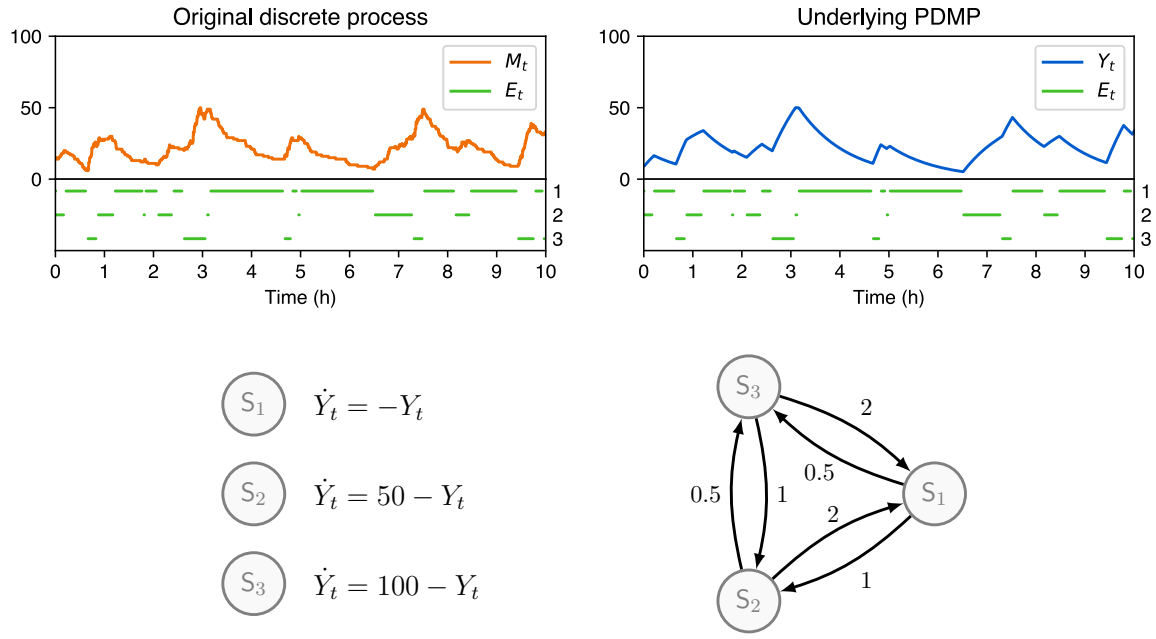


Figure 3.1 – Sample paths of the original process and the underlying piecewise-deterministic process for an example of multistate promoter: $n = 3$, $u = (0, 50, 100)$, $r_{2,1} = r_{3,1} = 2$, $r_{1,2} = r_{3,2} = 1$ and $r_{1,3} = r_{2,3} = 0.5$. Here the paths of (E_t, M_t) and (E_t, Y_t) are generated using the same path of E_t.

which is the adjoint of

$$\widetilde{\mathcal{L}} f = (C - xI)\partial_x f + Qf. \tag{3.10}$$

A comparison of $\widetilde{\mathcal{L}}$ with (3.2) reveals that we made significant progress by going from a discrete to a continuous description. Notably, we thereby obtain the generator of a process $(E_t, Y_t)_{t \geqslant 0}$ that is a typical PDMP of state space $[\![1, n]\!] \times \mathbb{R}_+$, with E_t being the same as before and Y_t following the (random) differential equation

$$\dot{Y}_t = \overline{u}(E_t) - Y_t \tag{3.11}$$

where $\overline{u}(i) = u_i$ for $i \in [\![1, n]\!]$. In other words, given the promoter state, the continuous variable Y_t follows the traditional *deterministic mass-action* kinetics of M in system (3.1): an example of such underlying PDMP is shown in Figure 3.1. The mixing distribution thus evolves exactly as we would expect when considering $[\mathsf{M}]$ continuous while keeping the $[\mathsf{S}_i]$ discrete, and indeed (3.8) can be rewritten as a simple system of coupled transport equations

$$\partial_t \mu + \partial_x[(C - xI)\mu] = H\mu$$

for which (3.11) is the characteristic curve corresponding to each vector component.

Although the previous method is only heuristic (boundary terms in the integration by parts may indeed not vanish), it is possible to get the same outcome rigorously using a dual approach related to Remark 3.3. More precisely, if $p(t)$ is the distribution of (E_t, M_t) and $\mu(t)$ is the distribution of (E_t, Y_t), then from the definition (3.3) of Ω, the generating function $g(z, t) = G_{p(t)}(z)$ satisfies the evolution equation

$$\partial_t g(z, t) + (z - 1)\partial_z g(z, t) = ((z - 1)C + H)g(z, t) \tag{3.12}$$

while from the definition (3.9) of $\widetilde{\Omega}$, the Laplace transform $\phi(s,t) = L_{\mu(t)}(s)$ follows

$$\partial_t \phi(s,t) + s\partial_s \phi(s,t) = (sC + H)\phi(s,t). \tag{3.13}$$

Clearly, equations (3.12) and (3.13) perfectly coincide up to the change of variable $s = z - 1$. As a result (see appendix 3.B.2 for details), the dynamics of $p(t)$ coincide on the space of Poisson mixtures with the dynamics of $\mu(t)$ in the following sense.

Proposition 3.4. *Let $(S_t)_{t\geqslant 0}$ and $(\widetilde{S}_t)_{t\geqslant 0}$ be the operator semigroups generated by Ω and $\widetilde{\Omega}$, that is, for any $p_0 \in \mathcal{M}_n(\mathbb{N})$ and $\mu_0 \in \mathcal{M}_n(\mathbb{R}_+)$:*

— *$p(t) = S_t p_0$ is the solution of (3.4) with initial condition $p(0) = p_0$;*

— *$\mu(t) = \widetilde{S}_t \mu_0$ is the solution of (3.8) with initial condition $\mu(0) = \mu_0$.*

Then for all $t \geqslant 0$, the space of Poisson mixtures $\mathcal{P}_n \subset \mathcal{M}_n(\mathbb{N})$ is an invariant subspace of S_t and we have the following commutative diagram:

$$\begin{array}{ccc} \mathcal{M}_n(\mathbb{R}_+) & \xrightarrow{\widetilde{S}_t} & \mathcal{M}_n(\mathbb{R}_+) \\ \downarrow P & & \downarrow P \\ \mathcal{P}_n & \xrightarrow{S_t} & \mathcal{P}_n \end{array}$$

that is, $P^{-1} S_t P = \widetilde{S}_t$ where P^{-1} is well-defined on $\mathcal{P}_n$.

It is worth noticing that since $(E_t, M_t)_{t\geqslant 0}$ is ergodic (Peccoud et Ycart, 1995), its unique stationary distribution belongs to the space $\mathcal{P}_n$ which is therefore clearly a fundamental invariant subspace. It is not that common to know such a nontrivial subspace when dealing with infinite-dimensional semigroups, and this interesting result strongly suggests the introduction of the underlying process $(E_t, Y_t)_{t\geqslant 0}$. Unsurprisingly, it is also ergodic (Benaïm *et al.*, 2012) so the same Poisson representation holds in stationary regime.

Corollary 3.5. *The stationary distribution of the original process $(E_t, M_t)_{t\geqslant 0}$ is the Poisson mixture $p = P\mu$ where μ is that of $(E_t, Y_t)_{t\geqslant 0}$.*

Analogs of Proposition 3.4 may be derived for any chemical system as a general consequence of the stochastic mass-action assumption but the resulting semigroups typically do not correspond to Markov processes, that is, they do not have an actual probabilistic interpretation (see Gardiner et Chaturvedi, 1977 for an interesting discussion). Here we obtain a well-defined process by letting E_t unchanged, so our approach is rather a "hybrid" Poisson representation. In our case, it is even possible to obtain a stronger result describing not only distributions but also sample paths: this approach appears in (Dattani et Barahona, 2017) but we slightly adapt it here to the "invariant subspace" viewpoint.

3.2.3 Path-based approach

In line with the chemical system (3.1) and noticing that $(E_t)_{t\geqslant 0}$ is itself a jump Markov process with generator Q, one may alternatively consider $(M_t)_{t\geqslant 0}$ as a birth-death process in random environment $(E_t)_{t\geqslant 0}$, which can be described by a scalar, conditional master equation (see appendix 3.B.3). The conditional generating function of mRNA given a promoter path is then defined by

$$\overline{g}(z,t) = \mathbb{E}\left[z^{M_t} | (E_\tau)_{\tau\geqslant 0}\right] = \mathbb{E}\left[z^{M_t} | (E_\tau)_{\tau\in[0,t]}\right]$$

and it satisfies the following partial differential equation:

$$\partial_t \overline{g}(z,t) + (z-1)\partial_z \overline{g}(z,t) = (z-1)\overline{u}(E_t)\overline{g}(z,t). \tag{3.14}$$

This is just the analog of (3.12), but much easier to solve since it is now a scalar transport equation. Using the standard method of characteristics, we get the following result.

Proposition 3.6. *Let $y_0 \in \mathbb{R}_+$. If $\overline{g}(z,0) = \exp(y_0(z-1))$, then we have*

$$\overline{g}(z,t) = \exp(Y_t(z-1))$$

for all $t \geqslant 0$, where

$$Y_t = e^{-t}y_0 + \int_0^t \overline{u}(E_\tau)e^{\tau - t}d\tau$$

is the unique solution of the differential equation (3.11) *such that $Y_0 = y_0$.*

Such Y_t is well-defined since $t \mapsto \overline{u}(E_t)$ is piecewise constant, and we can construct $(E_t, M_t)_{t\geqslant 0}$ and $(E_t, Y_t)_{t\geqslant 0}$ using the same path of $(E_t)_{t\geqslant 0}$ as in Figure 3.1. In this case, if $M_0 \sim \text{Poisson}(y_0)$ is independent of E_0, then by Proposition 3.6,

$$\mathbb{P}_{M_t|(E_\tau)_{\tau\in[0,t]}} = \mathbb{P}_{M_t|Y_t}$$

and more specifically if Y_0 is independent of E_0 and $M_0|Y_0 \sim \text{Poisson}(Y_0)$, we get

$$\forall t \geqslant 0, \quad M_t|Y_t \sim \text{Poisson}(Y_t). \tag{3.15}$$

One must stay aware that the M_t are not independent, even conditionally on $(Y_t)_{t\geqslant 0}$. However, we also obtain $\mathbb{E}[(M_t)_{t\geqslant 0}|(E_t)_{t\geqslant 0}] = \mathbb{E}[(M_t)_{t\geqslant 0}|(Y_t)_{t\geqslant 0}] = (Y_t)_{t\geqslant 0}$ and thus, as clearly perceptible in Figure 3.1, the whole path $(M_t)_{t\geqslant 0}$ can be interpreted as small Poisson-type fluctuations around $(Y_t)_{t\geqslant 0}$ which itself describes the core part of the dynamics. This link between the two processes is in fact not specific to our choice of promoter dynamics: see Dattani et Barahona (2017) for a more general presentation.

3.3 Underlying multivariate structure

Following the Poisson representation, our interest is now the process $(Y_t)_{t\geqslant 0}$ defined by (3.11). In this section, we slightly change our point of view in order to reveal interesting symmetries. More precisely, we introduce a multivariate process $(X_t)_{t\geqslant 0} = (X_{1,t}, \dots, X_{n,t})_{t\geqslant 0}$ with state space $\mathbb{R}_+^n = (\mathbb{R}_+)^n$, such that $X_0 \in \Delta^{n-1}$, the $(n-1)$-simplex defined by

$$\Delta^{n-1} = \left\{x \in \mathbb{R}_+^n \,|\, x_1 + \cdots + x_n = 1\right\},$$

and built from $(E_t)_{t\geqslant 0}$ so that when $E_t = i$,

$$\dot{X}_{j,t} = -X_{j,t} \;\text{ for } j \neq i \quad \text{and} \quad \dot{X}_{i,t} = \sum_{j\neq i} X_{j,t}. \tag{3.16}$$

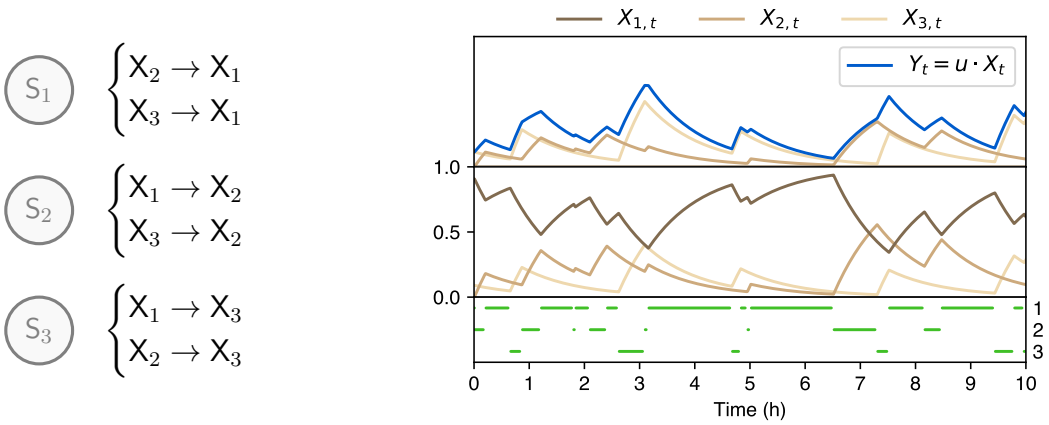


Figure 3.2 – Multivariate PDMP in the three-state case. Left: chemical reactions associated with each promoter state. Right: sample path corresponding to the promoter example of Figure 3.1, based on the same path of E_t. Using the same $u = (0, 50, 100)$ leads to the previous path of $Y_t = u \cdot X_t$.

Regarding the original chemical formulation (3.1), these equations simply correspond to deterministic mass-action kinetics of species $\mathsf{X}_1, \dots, \mathsf{X}_n$ following reactions

$$\mathsf{S}_i + \mathsf{X}_j \xrightarrow{d_0} \mathsf{S}_i + \mathsf{X}_i \quad \text{for } i, j \in [\![1, n]\!],\ i \neq j,$$

with $d_0 = 1$ here. The case $n = 3$ is shown in Figure 3.2, with a sample path of $(E_t, X_t)_{t \geqslant 0}$ based on the same $(E_t)_{t \geqslant 0}$ as in Figure 3.1. In particular, it is easy to check from (3.16) that $X_{1,t} + \cdots + X_{n,t}$ is conserved, and thus $X_t \in \Delta^{n-1}$ for all $t \geqslant 0$. The main point of introducing X_t is the following result.

Proposition 3.7. *Let $u = (u_1, \dots, u_n) \in \mathbb{R}_+^n$ denote the mRNA creation rates. Then for all $t \geqslant 0$,*

$$M_t | X_t \sim \text{Poisson}(u \cdot X_t) \tag{3.17}$$

where $u \cdot X_t = u_1 X_{1,t} + \cdots + u_n X_{n,t}$, whenever $M_0 | X_0 \sim \text{Poisson}(u \cdot X_0)$.

Proof. Following (3.15), we just need to show that $Y_t = u \cdot X_t$ whenever $Y_0 = u \cdot X_0$. When $E_t = i$ and since u is constant, the time derivative of $u \cdot X_t$ is

$$u_i \dot{X}_{i,t} + \sum_{j \neq i} u_j \dot{X}_{j,t} = u_i \sum_{j \neq i} X_{j,t} - \sum_{j \neq i} u_j X_{j,t} = u_i - u \cdot X_t$$

where we used (3.16) and the fact that $X_{1,t} + \cdots + X_{n,t} = 1$. The result immediately follows as Y_t satisfies the same differential equation, that is, $\dot{Y}_t = u_i - Y_t$. □

Interestingly, the representation (3.17) can be interpreted as Y_t being a projection of X_t on $\mathbb{R}_+$ using u. The initial condition in Proposition 3.7 turns out to be equivalent to $Y_0 \in [\min(u), \max(u)]$ in (3.15), which in fact is the physically relevant state space regarding the dynamics of Y_t (i.e., values taken by $[\mathsf{M}]$ when treated as a concentration) so it is not too restrictive. Note that by Corollary 3.5, the stationary distribution of $(M_t)_{t \geqslant 0}$ can always be represented as in (3.17) using that of $(X_t)_{t \geqslant 0}$.

Motivated by Proposition 3.7, we shall focus on $(E_t, X_t)_{t \geqslant 0}$ which will be referred to as the "multivariate PDMP" as it is clearly also a piecewise-deterministic Markov process. Its

generator is given from (3.16) by

$$\widetilde{\mathcal{L}}_n f(x) = Qf(x) + \sum_{i=1}^{n} F_i(x)\partial_{x_i} f(x) \tag{3.18}$$

for $x \in \mathbb{R}_+^n$, with $F_i(x) = (x_1 + \cdots + x_n)\,\mathrm{Diag}(e_i) - x_i I$ where $\mathrm{Diag}(e_i) \in \mathbb{R}^{n\times n}$ is the matrix whose only nonzero entry is $[\mathrm{Diag}(e_i)]_{i,i} = 1$.

3.3.1 Multivariate Laplace transform

Let $\mathcal{M}(\mathbb{R}_+^n)$ denote the space of finite measures on $\mathbb{R}_+^n$ and let $\mathcal{M}_n(\mathbb{R}_+^n) = \mathcal{M}(\mathbb{R}_+^n)^n$ represent finite measures on $[\![1,n]\!] \times \mathbb{R}_+^n$. In the remainder of this chapter, we consider a random variable

$$(E, X) \sim \mu$$

where $\mu = (\mu_1, \ldots, \mu_n)^\top \in \mathcal{M}_n(\mathbb{R}_+^n)$ is the stationary distribution of the multivariate PDMP. In line with our previous notation, we define the Laplace transform $\phi = L_\mu$ by

$$\phi_i(s) = \int_{\mathbb{R}_+^n} e^{s\cdot x} d\mu_i(x) = \mathbb{E}\left[\mathbb{1}_{\{E=i\}} e^{s\cdot X}\right], \quad \forall s \in (-\infty, 0]^n. \tag{3.19}$$

From (3.18) and the fact that $X \in \Delta^{n-1}$ (which is equivalent to $\partial_{s_1}\phi + \cdots + \partial_{s_n}\phi = \phi$), we find that ϕ satisfies

$$\sum_{i=1}^{n} s_i \partial_{s_i}\phi(s) = (D(s) + H)\phi(s) \tag{3.20}$$

with $D(s) = \mathrm{diag}(s_1, \ldots, s_n)$.

Besides, an analog of Remark 3.1 implies that ϕ can be extended to an entire function on $\mathbb{R}^n$. More precisely, we have

$$\phi(s) = \sum_{\alpha\in\mathbb{N}^n} m(\alpha)\frac{s_1^{\alpha_1}\cdots s_n^{\alpha_n}}{\alpha_1!\cdots\alpha_n!} \tag{3.21}$$

where $m(\alpha)$, using notation $\partial_s^\alpha = \partial_{s_1}^{\alpha_1}\cdots\partial_{s_n}^{\alpha_n}$ and $X^\alpha = {X_1}^{\alpha_1}\cdots{X_n}^{\alpha_n}$, is given by

$$m_i(\alpha) = \partial_s^\alpha \phi_i(0) = \mathbb{E}\left[\mathbb{1}_{\{E=i\}} X^\alpha\right]. \tag{3.22}$$

The convergence of the series (3.21) for all $s \in \mathbb{R}^n$ is then immediate as $m_i(\alpha) \in [0,1]$ for all $\alpha \in \mathbb{N}^n$, by (3.22) and the fact that $X \in \Delta^{n-1} \subset [0,1]^n$.

3.3.2 General recursion formula

We are now interested in solving system (3.20). Given some fixed $s \in \mathbb{R}^n$, the change of unknown $\Phi_s(\omega) = \phi(\omega s_1, \ldots, \omega s_n) = \phi(\omega s)$ for $\omega \in \mathbb{R}$ directly leads to a much simpler one-variable problem.

Proposition 3.8. *For all $s \in \mathbb{R}^n$, $\omega \mapsto \Phi_s(\omega)$ is solution of*

$$\omega\frac{d\Phi_s}{d\omega}(\omega) = (\omega D(s) + H)\Phi_s(\omega) \tag{3.23}$$

which is an ordinary differential system.

It should be noted from (3.19) that Φ_s is in fact the Laplace transform of (E, Y) where $Y = s \cdot X$ (i.e., $\Phi_s = L_{\overline{\mu}}$ with $(E, Y) \sim \overline{\mu}$). Combined with Lemma 3.2, this explains why (3.23) happens to coincide, in the special case $s = u$, with the main equation considered in (Innocentini *et al.*, 2013). A important consequence of (3.21) is that Φ_s can be expressed as a power series in ω, and some simple computation from (3.23) then leads to the following recurrence relation between the coefficients.

Corollary 3.9. *Let $s \in \mathbb{R}^n$. Then $\Phi_s(\omega) = \sum_{k=0}^{\infty} c_k(s)\omega^k$ where*

$$Hc_0(s) = 0 \quad \textit{and} \quad \forall k \geqslant 1, \quad c_k(s) = (kI - H)^{-1} D(s) c_{k-1}(s). \tag{3.24}$$

This recursion formula corresponds to the one used in (Innocentini *et al.*, 2013). The irreducibility of H is crucial here: a classic application of the Perron-Frobenius theorem [1] shows that eigenvalues of H all have negative real parts except the simple eigenvalue 0, so the recurrence (3.24) is well-defined and $(c_k(s))_{k \geqslant 0}$ is unique up to a multiplicative constant. Taking $\omega = 1$, we finally obtain

$$\phi(s) = \sum_{k=0}^{\infty} c_k(s) \tag{3.25}$$

which can be seen as a particular choice of summation in (3.21). Since the distribution of E is by definition $c_0(s) = c_0 = m(0)$, the sequence $(c_k(s))_{k \geqslant 0}$ is unique, confirming the uniqueness of μ under the assumption that $X_0 \in \Delta^{n-1}$ almost surely.

Although useful for numerical computation, especially when H is diagonalizable, the recurrence relation (3.24) does not make ϕ really explicit, the main challenge being that matrices $D(s)$ and H do not commute. Remarkably, the case where only one s_i is nonzero turns out to be explicitly solvable: it is the object of section 3.4. Let us however present a fully solvable configuration in the next subsection.

3.3.3 A fully solvable case

Let $\alpha_1, \ldots, \alpha_n$ be positive parameters and consider the very particular case

$$r_{i,j} = \alpha_j, \quad \forall i, j \in [\![1, n]\!],\ i \neq j. \tag{3.26}$$

Namely, each promoter state S_i can be reached directly from all other states S_j with the same rate α_i. As an example, the three-state promoter in Figure 3.1 and thus the multivariate sample path in Figure 3.2 belong to this case. Such dynamics clearly have no memory: although simplistic from a biological perspective, this situation may be viewed as a useful first step, giving some interesting insight into μ in general.

Proposition 3.10. *If the promoter transition rates satisfy* (3.26)*, the stationary distribution of mRNA coincides with the hierarchical model*

$$\begin{aligned} X &\sim \text{Dirichlet}(\alpha_1, \ldots, \alpha_n) \\ M|X &\sim \text{Poisson}\left(u_1 X_1 + \cdots + u_n X_n\right) \end{aligned}$$

where X corresponds to the multivariate PDMP and M corresponds to mRNA.

1. To the irreducible nonnegative matrix $\frac{1}{\alpha} H + I$ where $\alpha = \max_i \left\{ \sum_{j \neq i} r_{i,j} \right\} > 0$ (see Appendix 3.C).

Proof. In this easy case it is possible to use Corollary 3.9 but not even necessary. Indeed, we obtain from (3.18) that μ satisfies the stationary equation

$$\sum_{i=1}^{n} \partial_{x_i}(F_i\mu) = H\mu \tag{3.27}$$

with $F_i(x) = (x_1 + \cdots + x_n)\,\mathrm{Diag}(e_i) - x_i I$ and H derived from (3.26), that is:

$$\sum_{j\neq i} \partial_{x_i}(x_j\mu_i) - \partial_{x_j}(x_j\mu_i) = \alpha_i \sum_{j\neq i} \mu_j - \mu_i \sum_{j\neq i} \alpha_j, \quad \forall i \in [\![1, n]\!].$$

This system turns out to be solvable "piece by piece". Let σ denote the induced Lebesgue measure on Δ^{n-1}, and let $\mu \in \mathcal{M}_n(\mathbb{R}_+^n)$ be defined by the density $f = (f_1, \ldots, f_n)^\top$ with respect to σ (meaning μ has all its mass in Δ^{n-1}) up to a normalizing constant, where

$$f_i(x) = x_i \prod_{j=1}^{n} x_j^{\alpha_j - 1}, \quad \forall i \in [\![1, n]\!].$$

Then we easily get $\partial_{x_i}(x_j f_i) = \alpha_i f_j$ and $\partial_{x_j}(x_j f_i) = \alpha_j f_i$ for all $i, j \in [\![1, n]\!]$, $i \neq j$, and it follows that μ is solution of (3.27) in the weak sense. In other words, we have

$$\mathbb{P}_{X|E=i} = \mathrm{Dirichlet}(\alpha_1, \ldots, \alpha_{i-1}, \alpha_i + 1, \alpha_{i+1}, \ldots, \alpha_n), \quad \forall i \in [\![1, n]\!]$$

and marginalizing over E leads to $\mathbb{P}_X = \mu_1 + \cdots + \mu_n = \mathrm{Dirichlet}(\alpha_1, \ldots, \alpha_n)$. □

3.4 Complete solution for refractory promoters

Recall that a promoter state $i \in [\![1, n]\!]$ is *active* if $u_i > 0$ and *inactive* if $u_i = 0$. In this section we consider the particular case of *refractory promoters*, that is, for which only one state is active. In line with the previous section, we consider a random variable M generated by

$$M|X \sim \mathrm{Poisson}(u \cdot X)$$

so $\mathbb{P}_M$ is the mRNA stationary distribution. Without loss of generality, we assume that the active state is the first one (Figure 3.3) with $u_1 = \nu > 0$ so that $u \cdot X = \nu X_1$. We derive an explicit formula for $\mathbb{P}_M$ in this case, thereby simplifying and generalizing the results in (Zhou *et al.*, 2012), and extend some of the ideas in (Dattani, 2016; Dattani et Barahona, 2017) concerning X_1.

Remark 3.11. It should now be clear that refractory promoters are associated with marginal distributions of X. For instance, one easily recovers the fact that $\mathbb{P}_M$ is a scaled Beta-Poisson mixture in the case of the two-state model (Kim et Marioni, 2013): when $n = 2$, $r_{2,1} = \alpha_1$ and $r_{1,2} = \alpha_2$, we get $X \sim \mathrm{Dirichlet}(\alpha_1, \alpha_2)$ by Proposition 3.10 and then it is well-known that $X_1 \sim \mathrm{Beta}(\alpha_1, \alpha_2)$ and $X_2 \sim \mathrm{Beta}(\alpha_2, \alpha_1)$.

3.4.1 Notation and definitions

The refractory case also provides notions of active and inactive *periods*, which are respectively the time T_1 spent in the active state before getting inactive and the time T_0 spent in

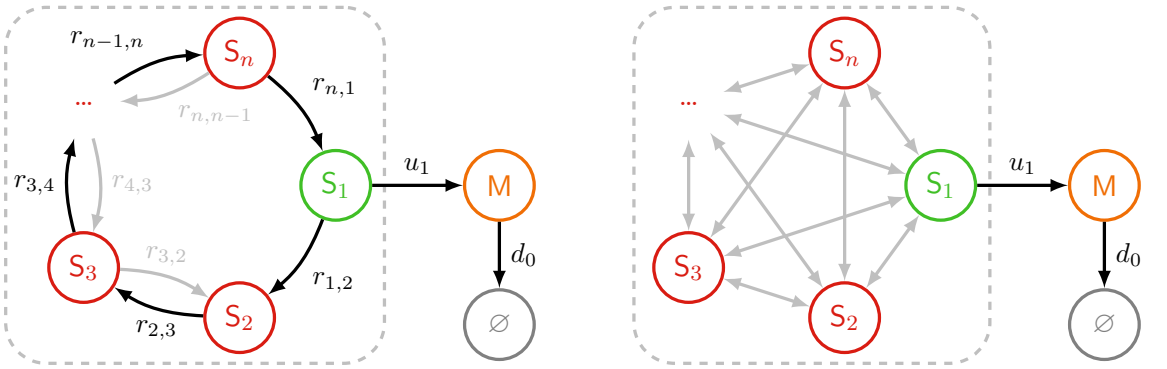


Figure 3.3 – Dynamics of the system in the case of refractory promoters. Black arrows correspond to reactions with positive rates while grey ones indicate reactions that can have rate 0. Left: cyclic refractory promoters as considered by Zhou *et al.* (2012). Right: general refractory promoters as considered here.

other states before getting active. By definition of the process, T_1 has density f_{T_1} on $[0, +\infty)$ given by $f_{T_1}(t) = \lambda \exp(-\lambda t)$ where $\lambda = \sum_{j \neq 1} r_{1,j}$, and the density of T_0 is also available explicitly.

Lemma 3.12. *The inactive period T_0 has density f_{T_0} defined on $[0, +\infty)$ by*

$$f_{T_0}(t) = \left[\tilde{H} \exp(t\tilde{H})\pi\right]_1$$

where $\pi = (\pi_1, \dots, \pi_n)^\top$ is defined by $\pi_1 = 0$ and $\pi_i = r_{1,i} / \sum_{j \neq 1} r_{1,j}$ for $i \neq 1$, and $\tilde{H} \in \mathbb{R}^{n \times n}$ is obtained from H by replacing the first column with zeros.

Proof. Consider the Markov process $(\tilde{E}_t)_{t \geqslant 0}$ with generator $\tilde{H}$ (i.e., it gets "stuck" in state $i = 1$ as soon as it is reached) and such that $\mathbb{P}_{\tilde{E}_0} = \pi$. The inactive period can then be defined as $T_0 = \inf\{t > 0 \,|\, \tilde{E}_t = 1\}$. Its cumulative distribution function is

$$F_{T_0}(t) = \mathbb{P}(T_0 \leqslant t) = \mathbb{P}(\tilde{E}_t = 1) = \left[\exp(t\tilde{H})\pi\right]_1$$

and the result follows by taking the derivative of F_{T_0}. □

The distribution of T_0 is not the main point here but it enlightens the underlying linear algebra that also appears in the next results. In addition, we shall use Lemma 3.12 in the Applications section to gain insight into the particular dynamics of some promoters.

As found in (Zhou *et al.*, 2012) and (Dattani et Barahona, 2017), distributions of M and X_1 can be expressed in a compact way using generalized hypergeometric functions. Let us introduce some related notation, borrowed from (Slater, 1966). Given $A \in \mathbb{N}$ and $a = (a_1, \dots, a_A) \in \mathbb{C}^A$, we define

$$(a)_k = \prod_{i=1}^{A} \frac{\Gamma(a_i + k)}{\Gamma(a_i)} = \prod_{i=1}^{A} a_i(a_i + 1) \cdots (a_i + k - 1)$$

for $k \in \mathbb{N}$, adopting the convention $(a)_k = 1$ if $A = 0$. Then, given also $B \in \mathbb{N}$ and

$b = (b_1, \dots, b_B) \in \mathbb{C}^B$, we define the hypergeometric function ${}_AF_B$ as

$$ {}_AF_B[(a);(b);x] = \sum_{k=0}^{\infty} \frac{(a)_k}{(b)_k} \frac{x^k}{k!} = \sum_{k=0}^{\infty} \frac{(a)_k}{(b)_k} \frac{x^k}{k!} $$

for $x \in \mathbb{R}$ such that the series is well-defined and converges. For $z \in \mathbb{C}$, we shall write $a + z = (a_1 + z, \dots, a_A + z) \in \mathbb{C}^A$ and $b + z = (b_1 + z, \dots, b_B + z) \in \mathbb{C}^B$.

We finally set $n = N + 1$ to simplify the notation in this section, so that $N \geqslant 1$ is the number of inactive states. Then, combining the Perron-Frobenius theorem as mentioned beside Corollary 3.9 with some other linear algebra results (see Appendix 3.C), we can define two fundamental families of eigenvalues.

Lemma 3.13. *For $i \in [\![1, n]\!]$, let $H_{\{i\}} \in \mathbb{R}^{N \times N}$ be the matrix obtained from H by removing the i-th row and the i-th column. Furthermore, let*

$$ a^{(i)} = (a_1^i, \dots, a_N^i) \in \mathbb{C}^N \quad \text{and} \quad b = (b_1, \dots, b_N) \in \mathbb{C}^N $$

respectively denote the eigenvalues of $-H_{\{i\}}$ and the nonzero eigenvalues of $-H$, counted with multiplicity. Then $a_1^i, \dots, a_N^i$, $b_1, \dots, b_N$ all have positive real parts and we have

$$ \mathbb{P}_E(i) = \prod_{k=1}^{N} \frac{a_k^i}{b_k}, \quad \forall i \in [\![1, n]\!] $$

where $\mathbb{P}_E$ is the promoter stationary distribution.

Remarkably, it turns out that $\mathbb{P}_M$ is parametrized by the eigenvalues of Lemma 3.13.

3.4.2 Exact mRNA distribution

Let us first characterize the continuous component X_1. In all that follows, we consider $a = a^{(1)}$ and b as defined in Lemma 3.13. The results are based on X_1 for simplicity but immediately generalize to any X_i by replacing $a = a^{(1)}$ with $a^{(i)}$.

Theorem 3.14. *The Laplace transform of X_1 is given by*

$$ L_{X_1}(s_1) = \mathbb{E}\left[e^{s_1 X_1}\right] = {}_NF_N[(a);(b);s_1]. \tag{3.28} $$

Proof. The idea is to solve the recurrence relation (3.24) to get $c_k(s)$ for all $k \in \mathbb{N}$, assuming that $s = (s_1, 0, \dots, 0)$. Since we marginalize over the promoter state E, we are only interested in $\bar{c}_k = c_{k,1} + \cdots + c_{k,n}$. First, summing vectorial components in $(kI - H)c_k(s) = D(s)c_{k-1}(s)$ yields $k\bar{c}_k(s) = s_1 c_{k-1,1}(s)$ so we just need $c_{k,1}(s)$. Second, it is straightforward to see that the $(1,1)$-cofactor of $kI - H$ is equal to $\det(kI - H_{\{1\}}) = (a_1 + k) \cdots (a_N + k)$ and we use it in (3.24) through the standard inversion formula to get the scalar recurrence relation

$$ c_{k,1}(s) = \frac{\det(kI - H_{\{1\}})}{\det(kI - H)} s_1 c_{k-1,1}(s) = \frac{(a_1 + k) \cdots (a_N + k)}{k(b_1 + k) \cdots (b_N + k)} s_1 c_{k-1,1}(s), \quad \forall k \geqslant 1. $$

The initial term is $c_{0,1} = \mathbb{P}_E(1) = (a_1 \cdots a_N)/(b_1 \cdots b_N)$ by Lemma 3.13. Finally,

$$ \bar{c}_k(s) = \frac{s_1}{k} c_{k-1,1}(s) = \frac{(a)_k}{(b)_k} \frac{s_1^k}{k!}, \quad \forall k \in \mathbb{N} $$

and the result immediately follows from (3.25). □

Computing the derivatives of $s_1 \mapsto L_{X_1}(\nu s_1)$ and using Lemma 3.2, we obtain the mRNA stationary distribution for general refractory promoters.

Corollary 3.15. *If $u = (\nu, 0, \dots, 0)$, the distribution of M is given by*

$$\mathbb{P}_M(k) = \frac{\nu^k}{k!}\frac{(a)_k}{(b)_k}{}_N F_N[(a+k);(b+k);-\nu]. \tag{3.29}$$

Note that since matrices $-H$ and $-H_{\{1\}}$ are real, their complex eigenvalues come by conjugate pairs so $(a+m)_k$ and $(b+m)_k$ for $m \in \mathbb{N}$ are always real numbers.

Remark 3.16. When ν is large, the Poisson layer becomes negligible, meaning $M \approx \nu X_1$, so we can alternatively use (3.29) to approximate the distribution of X_1. More precisely, if f_{X_1} denotes the density of X_1, we have

$$\nu \gg 1 \quad \Rightarrow \quad f_{X_1}(k/\nu) \approx \mathbb{P}_M(k)$$

which in fact corresponds to the Post-Widder inversion formula applied to L_{X_1}.

3.4.3 Density of the mixing distribution

When computing $\mathbb{P}_M$, it is common for tractability reasons to take a rather small value for the scale parameter ν, which is coherent since $\mathbb{P}_M(k)$ vanishes quickly for $k > \nu$ by definition of the Poisson mixture. However, quantitative biological experiments often suggest $\nu = 10^3$ or more (Albayrak *et al.*, 2016; Richard *et al.*, 2016). Equation (3.29) then corresponds as noted above to the Post-Widder inversion formula for the distribution of X_1, which emerges even more as the core part of $\mathbb{P}_M$.

It is therefore interesting to consider deriving the density f_{X_1} in exact form, that is, directly inverting the Laplace transform (3.28). Fortunately, one does not have to do this from scratch as L_{X_1} belongs to a well-known class (Slater, 1966). More precisely, the idea is to invert the *Mellin transform* of X_1, defined as the meromorphic function $M_{X_1}(z) = \mathbb{E}\left[X_1^{z-1}\right]$, which coincides with the moments of X_1 given from (3.28) by

$$\mathbb{E}\left[X_1^k\right] = \frac{(a)_k}{(b)_k} = \prod_{i=1}^{N}\frac{\Gamma(b_i)\Gamma(a_i+k)}{\Gamma(a_i)\Gamma(b_i+k)}, \quad \forall k \in \mathbb{N}. \tag{3.30}$$

It is possible to show using an extension of Carlson's theorem that the right-hand side of (3.30) actually defines M_{X_1} (replacing k with $z-1$), but here such a technical result is not needed since we know by (3.21) that the distribution of X_1 is characterized by its moments. Namely, we only need to find the unique distribution on $[0,1]$ with moments (3.30), and by Mellin inversion this one is defined by the density

$$f_{X_1}(x) = \frac{1}{2\pi\mathrm{i}}\prod_{i=1}^{N}\frac{\Gamma(b_i)}{\Gamma(a_i)}\int_{0-\mathrm{i}\infty}^{0+\mathrm{i}\infty}\prod_{i=1}^{N}\frac{\Gamma(a_i+z)}{\Gamma(b_i+z)}x^{-z-1}dz \tag{3.31}$$

for $x \in (0,1)$. Up to a multiplicative constant, this is a standard Meijer G-function (Springer et Thompson, 1970) and thus one can efficiently compute f_{X_1} using numerical packages such as `mpmath`. Furthermore, the following result provides an actual explicit form in most cases.

Theorem 3.17. *Assume that $a_i - a_j \notin \mathbb{Z}$ for all $i, j \in [\![1, N]\!]$, $i \neq j$. Then we have*

$$f_{X_1}(x) = \frac{1}{A} \sum_{i=1}^{N} B_i \, x^{a_i - 1} {}_N F_{N-1}[(1 + a_i - b); (1 - a_i'); x] \tag{3.32}$$

where $a_i' = (a_1, \ldots, a_{i-1}, a_{i+1}, \ldots, a_N) \in \mathbb{C}^{N-1}$,

$$A = \prod_{i=1}^{N} \frac{\Gamma(a_i)}{\Gamma(b_i)} \quad \textit{and} \quad B_i = \frac{\prod_{j \neq i} \Gamma(a_j - a_i)}{\prod_{j=1}^{N} \Gamma(b_j - a_i)}.$$

The proof simply consists in evaluating (3.31) by the method of residues and it appears in (Slater, 1966, p. 152, equation (4.8.1.16)) as a particular case. Note that the poles of the integrand (located at $z = -a_i - k$ for $k \in \mathbb{N}$) are simple by hypothesis. The case with multiple poles is treated extensively by Springer et Thompson (1970) but is more involved. When $a_i - b_j \in \mathbb{N}$, one simply has $B_i = 0$ so there is no restriction on b. Note also that `mpmath` appears to use (3.32) with a perturbation technique in the general case rather than performing complex integration in (3.31).

Remark 3.18. The "Dirichlet" promoter model (3.26) can indeed be recovered as a special case of (3.31) where most terms vanish: one a_i (say a_1) is equal to α_1 while all other eigenvalues (b_1 and a_i, b_i for $i \geqslant 2$) are equal to $\alpha_1 + \cdots + \alpha_n$ so $X_1 \sim \mathrm{Beta}(\alpha_1, \alpha_1 + \cdots + \alpha_n)$.

3.4.4 Beta-product distributions

Consider the case $n = 3$ with rates $r_{1,2} = 10$, $r_{2,3} = r_{3,1} = 2$, $r_{3,2} = 1$ and $r_{2,1} = r_{1,3} = 0$. This gives $a = (1, 4)$ and $b = (6, 9)$. Then it is easy to show using moments (3.30) that X_1 has the same distribution as $Z_1 Z_2$ where $(Z_1, Z_2) \sim \mathrm{Beta}(1, 5) \otimes \mathrm{Beta}(4, 5)$ or equivalently $(Z_1, Z_2) \sim \mathrm{Beta}(1, 8) \otimes \mathrm{Beta}(4, 2)$. More generally, if a and b are real with $a_i < b_i$ for all i, then X_1 can be interpreted as a product of independent Beta-distributed random variables (Springer et Thompson, 1970), namely,

$$X_1 = Z_1 \cdots Z_N \quad \text{with} \quad Z_i \sim \mathrm{Beta}(a_i, b_i - a_i).$$

Such Beta-product distributions also arise in statistics (Dufresne, 2010) and mathematical physics (Dunkl, 2013), and it is fruitful to extend them to complex a and b such that the moment sequence (3.30) stays real: this indeed generates "new" distributions that cannot be realized as Beta products with real parameters (Dufresne, 2010). From this viewpoint, the multivariate PDMP turns out to be a very natural way to generate, using real transition matrices, Beta-product distributions with complex parameters.

In the case $n = 3$, let us mention an alternative to Theorem 3.17 that is always valid (i.e., also when $a_1 - a_2 \in \mathbb{Z}$). Indeed, similarly to the real case (Dufresne, 2010; Dunkl, 2013) we get

$$f_{X_1}(x) = \frac{\Gamma(b_1)\Gamma(b_2)}{\Gamma(a_1)\Gamma(a_2)\Gamma(\delta)} x^{a_1 - 1} (1 - x)^{\delta - 1} {}_2F_1[(b_1 - a_2, b_2 - a_2); (\delta); 1 - x]$$

where $\delta = b_1 + b_2 - a_1 - a_2 = r_{1,2} + r_{1,3} > 0$. Note that a_1 and a_2 are always real in this case so the hypergeometric series has nonnegative coefficients, and thus f_{X_1} can be interpreted as a mixture of Beta distributions.

Applications

In this section we show some interesting examples of refractory promoters, still assuming that the active state is $i = 1$ and taking $\nu = 10^3$. The general formulas for distributions of T_0, M and X_1 were implemented in Python, making use of the `mpmath` package to compute (3.29) and (3.31).

3.4.5 Irreversible cyclic promoters

In this case, the promoter is progressing irreversibly through N inactive states (from 2 to $n = N + 1$) before reaching the active state 1, and so on (Figure 3.3). It is a straightforward generalization (Zoller *et al.*, 2015) of the two-state model, which corresponds to $N = 1$. As shown in Figure 3.4, increasing N while keeping $\mathbb{E}[T_0]$ fixed tends to decrease $\mathrm{Var}(T_0)$, which rather intuitively decreases $\mathrm{Var}(M)$.

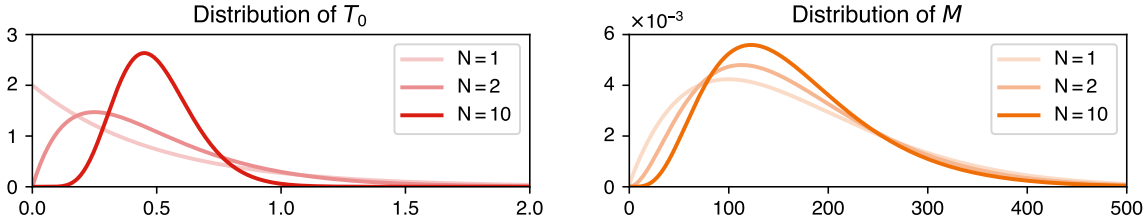


Figure 3.4 – Irreversible cyclic refractory promoters for $n = N + 1$ with $N \geqslant 1$ inactive states. Here we set $r_{1,2} = 10$ and $r_{i,i+1} = r_{n,1} = 2N$ for $i \in [\![2, N]\!]$, other rates being zero, so $T_0 \sim \mathrm{Gamma}(N, 2N)$. For $N = 10$, the distribution of M is close to its limit when $N \to \infty$, corresponding to $T_0 = 1/2$.

3.4.6 Irreversible cycle with a shortcut

We now consider a more complex inactive period in Figure 3.5, characterized by a five-state cycle with a "shortcut" from state 1 to state 4. The rates are chosen so that the promoter follows the long cycle most of the time, whereas it can sometimes bypass states 2 and 3, leading to a bimodal distribution for T_0. However, the two modes are not easily detectable in sample paths and the result is indeed a unimodal distribution for M.

3.4.7 Multiple cycles

Finally, Figure 3.6 shows a promoter with two distinct cycles, which leads to a bimodal distribution for M. This time one can see two typical inactive periods in sample paths, but T_0 appears unimodal: in fact the long cycle is rare compared to the short one so the corresponding mass is flattened.

Discussion and perspectives

In this work we derived an explicit Poisson representation for a standard model of gene expression, based on a multistate promoter. Compared to the original (Gardiner et Chaturvedi, 1977), our approach is hybrid in that we only represent M as a Poisson mixture while keeping a basic description of S_i. The underlying dynamics then correspond to a Markov process,

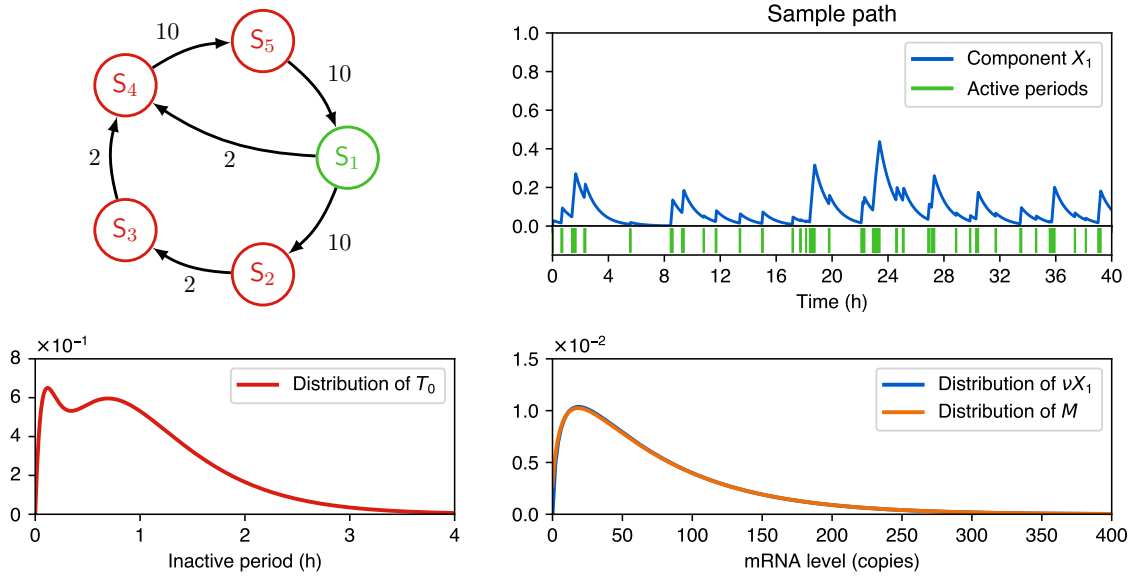


Figure 3.5 – Example of refractory promoter with a bimodal inactive period.

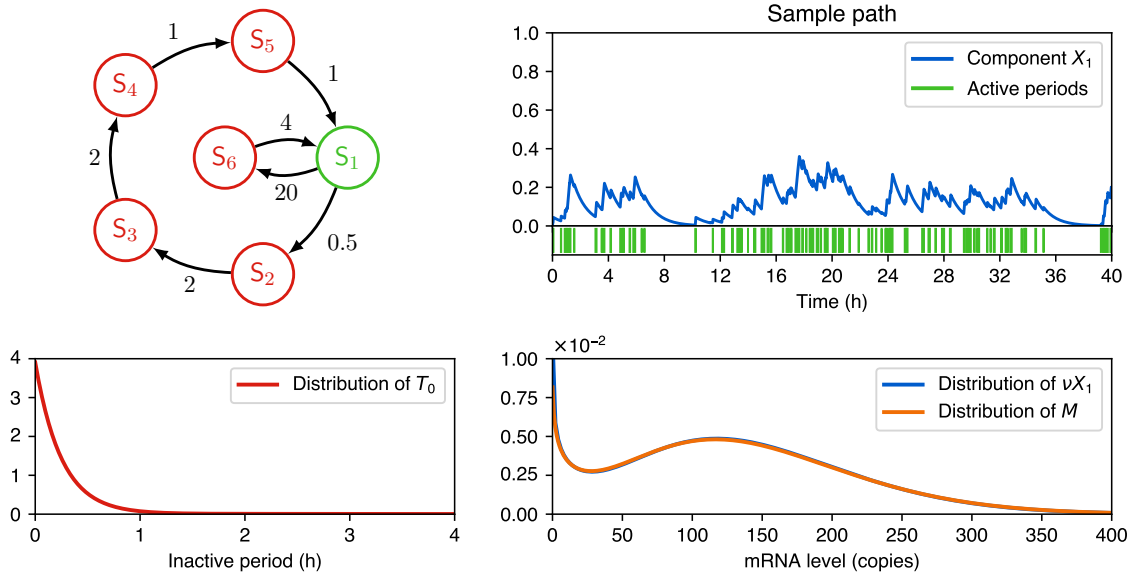


Figure 3.6 – Example of refractory promoter leading to a bimodal mRNA distribution.

which is not the case in general with reactions involving or producing more than one molecule (e.g., $\mathsf{S}_i \to \mathsf{S}_i + \mathsf{M}$). The explicit form of the mRNA distribution in the refractory case is similar to that of (Dattani, 2016; Zhou *et al.*, 2012), but we used the underlying linear structure rather that exploiting particular promoter transitions. This led to more general results with simpler proofs, and also enabled to identify marginals of the underlying multivariate PDMP as extending the class of beta-product distributions.

The PDMP viewpoint is itself getting well-established in biological applications because of its great [modeling power]/[mathematical complexity] ratio (Rudnicki et Tyran-Kamińska, 2017). In fact, it is relevant and already used in various situations outside biology, for example in the so-called fluid queuing theory where the two-state model also has a meaning (Boxma *et al.*, 2005). Figures 3.5 and 3.6 show that in biologically realistic conditions (Albayrak *et al.*, 2016; Singer *et al.*, 2014), the distribution of M efficiently approximates the one of νX_1, or in other words the Poisson layer adds a very small amount of noise to the PMDP layer. Besides, the mRNA bimodality in Figure 3.6 is exactly the one observed in practice (Singer *et al.*, 2014), that is, with a gamma-like tail. We emphasize that such multimodality differs from the one considered in (Innocentini *et al.*, 2013), which comes from long stays in distinct active states (i.e., with different u_i values) and has a much shorter, poissonian tail (see Albayrak *et al.*, 2016 for a quantitative illustration). In particular, contrary to a somewhat widespread belief, the two-state model is absolutely unable to reproduce the gamma-like bimodality. The promoter structure of Figure 3.6 gets around this by generating two latent "bursting frequencies", but it might be better to let such frequencies emerge from actual gene networks (i.e., coupled gene expression models) such as the well-known toggle-switch pattern (Herbach *et al.*, 2017).

Having Proposition 3.10 in mind, it is clear that vectors a and b are in general not identifiable from the mRNA distribution, as Dirichlet marginals are Beta and thus indistinguishable from the two-state model. In practice, Figures 3.4 to 3.6 suggest that distributions of M or νX_1 in the bursty regime may be reasonably approximated by gamma or mixtures of gamma distributions. This favors the two-state model as a relevant approximation in many cases since the gamma distribution is nothing but the bursty limit (i.e., $r_{1,2} \gg r_{2,1}$ if the active state is $i = 1$) of this model (Herbach *et al.*, 2017).

We could not find a general explicit form for the joint density of the multivariate PDMP, but the Dirichlet case is well-known to have its Laplace transform corresponding to a Lauricella hypergeometric series. Although the general case might be much more involved, one can hope for a nice form since marginals are tractable. Intuitively, the difficulty comes from the dependence between components $X_1, \ldots, X_n$, which happens to be trivial for the Dirichlet distribution (reduced to $X_1 + \cdots + X_n = 1$). Knowing the general joint distribution would be interesting not only mathematically, but also from a biological point of view as it would enable to describe further complexity layers. Indeed, the translation stage is commonly modeled by

$$\mathsf{M} \xrightarrow{v} \mathsf{M} + \mathsf{P}, \quad \mathsf{P} \xrightarrow{d_1} \varnothing$$

where P is the translated protein, and clearly this stage can be viewed for P exactly as the transcription stage with respect to M. Hence, the multivariate PDMP approach could hopefully give some useful insight for deriving the exact protein distribution, which is known to be a very difficult problem.

Appendix 3.A – Spaces

In this section we provide some details about function spaces and operators used in the chapter. First, $\mathcal{M}(\mathbb{R}_+)$ and $\mathcal{M}(\mathbb{N})$ respectively denote measures on $(\mathbb{R}_+, \mathcal{B}(\mathbb{R}_+))$ and $(\mathbb{N}, \mathcal{P}(\mathbb{N}))$, where $\mathcal{B}(\mathbb{R}_+)$ is the Borel algebra over $\mathbb{R}_+$ and $\mathcal{P}(\mathbb{N})$ is the power set of $\mathbb{N}$. The first space is equipped with the total variation norm, defined by

$$\|\mu\| = \sup\{\mu(A) - \mu(\mathbb{R}_+ \backslash A) \mid A \in \mathcal{B}(\mathbb{R}_+)\} = \mu^+(\mathbb{R}_+) + \mu^-(\mathbb{R}_+)$$

where $\mu = \mu^+ - \mu^-$ is the Jordan decomposition of $\mu \in \mathcal{M}(\mathbb{R}_+)$. The total variation norm is defined similarly on $\mathcal{M}(\mathbb{N})$, the Jordan decomposition being trivial in this case. Furthermore, the Riesz-Markov representation theorem identifies $\mathcal{M}(\mathbb{R}_+)$ and $\mathcal{M}(\mathbb{N})$ as the duals of Banach spaces $(\mathcal{C}_0(\mathbb{R}_+), \|\cdot\|_\infty)$ and $(c_0(\mathbb{N}), \|\cdot\|_\infty)$, respectively, where $\mathcal{C}_0(\mathbb{R}_+)$ is the space of continuous functions on $\mathbb{R}_+$ converging to zero at $+\infty$, and $c_0(\mathbb{N})$ is the space of real sequences converging to zero. Remarkably, $\|\cdot\|$ then corresponds to the dual norm so $(\mathcal{M}(\mathbb{R}_+), \|\cdot\|)$ and $(\mathcal{M}(\mathbb{N}), \|\cdot\|)$ are Banach spaces. Finally, all of this extends to $\mathcal{M}_n(\mathbb{R}_+)$ and $\mathcal{M}_n(\mathbb{N})$ as defined in section 3.2.1 (say, with $\|\mu\| = \|\mu_1\| + \cdots + \|\mu_n\|$ for $\mu \in \mathcal{M}_n(\mathbb{R}_+)$ or $\mathcal{M}_n(\mathbb{N})$), which are duals of

$$\mathcal{C}_0^n = \mathcal{C}_0([\![1, n]\!] \times \mathbb{R}_+) = \mathcal{C}_0(\mathbb{R}_+)^n \quad \text{and} \quad c_0^n = c_0([\![1, n]\!] \times \mathbb{N}) = c_0(\mathbb{N})^n.$$

We now introduce the generator $\mathcal{L}$ of the jump Markov process $(E_t, M_t)_{t \geqslant 0}$ as the infinitesimal generator of the semigroup of bounded operators $T_t : c_0^n \to c_0^n$ defined by

$$T_t f(i,k) = \mathbb{E}[f(E_t, M_t) | E_0 = i, M_0 = k], \quad \forall (i,k) \in [\![1, n]\!] \times \mathbb{N}$$

for $t \geqslant 0$ and $f \in c_0^n$. Similarly, the generator $\widetilde{\mathcal{L}}$ of the piecewise-deterministic Markov process $(E_t, Y_t)_{t \geqslant 0}$ is the infinitesimal generator of $\widetilde{T}_t : \mathcal{C}_0^n \to \mathcal{C}_0^n$ defined by

$$\widetilde{T}_t f(i,x) = \mathbb{E}[f(E_t, Y_t) | E_0 = i, Y_0 = x], \quad \forall (i,x) \in [\![1, n]\!] \times \mathbb{R}_+$$

for $t \geqslant 0$ and $f \in \mathcal{C}_0^n$. Note that $(T_t)_{t \geqslant 0}$ and $(\widetilde{T}_t)_{t \geqslant 0}$ are indeed (strongly continuous) semigroups and that $\mathcal{L}$ and $\widetilde{\mathcal{L}}$ coincide with (3.2) and (3.10): this is standard and follows from construction of the processes. Besides, we do not need the precise domains of the generators, but only subspaces that are dense in $(c_0^n, \|\cdot\|_\infty)$ and $(\mathcal{C}_0^n, \|\cdot\|_\infty)$: one can choose sequences that have finitely many nonzero elements and restrictions to $\mathbb{R}_+$ of compactly-supported smooth functions on $\mathbb{R}$, respectively.

As a result, the standard semigroup theory directly applies and the Kolmogorov backward equations of $(E_t, M_t)_{t \geqslant 0}$ and $(E_t, Y_t)_{t \geqslant 0}$ can be defined as well-posed Cauchy problems (Pazy, 1983). Let us now discuss the forward equations (3.4) and (3.8), which are the main point of Proposition 3.4. Our choice here is to directly consider the well-defined adjoint semigroups $S_t = T_t^*$ and $\widetilde{S}_t = \widetilde{T}_t^*$ rather than attempting at a precise definition of the forward Cauchy problems, whose solutions should by definition be based on these adjoint semigroups anyway.

Remark 3.19. The semigroup $S_t : \mathcal{M}_n(\mathbb{N}) \to \mathcal{M}_n(\mathbb{N})$ is indeed strongly continuous with generator Ω as in (3.3) but $\widetilde{S}_t : \mathcal{M}_n(\mathbb{R}_+) \to \mathcal{M}_n(\mathbb{R}_+)$ is *not*: the domain of its generator $\widetilde{\Omega}$ is not dense in $(\mathcal{M}_n(\mathbb{R}_+), \|\cdot\|)$, so there is no hope for a dense subspace on which it could be defined "strongly". To avoid this, a typical option is to embed $\mathcal{M}_n(\mathbb{R}_+)$ in a larger space of generalized functions and define $\widetilde{\Omega}$ in a weak sense, as done implicitly in (3.9). It is also

possible (Rudnicki et Tyran-Kamińska, 2017) to consider the subspace $L^1(\mathbb{R}_+)^n$, on which $\|\cdot\|$ coincides with $\|\cdot\|_1$ so that $\widetilde{\Omega}$ can be densely defined on smooth functions. Alternatively, we conjecture that one could get a strongly continuous semigroup using the Kantorovich-Rubinstein norm (see Hille et Worm, 2009). This is beyond the scope of this work but would be a very natural choice for applying the (forward) semigroup theory to PDMPs in general (e.g., see Benaïm *et al.*, 2012; Malrieu, 2015).

Appendix 3.B – Deriving the Poisson representation

3.B.1 Ansatz-based approach

This is the direct but non rigorous way to derive the operator $\widetilde{\Omega}$. Let $f = (f_1, \dots, f_n)^\top$ be a smooth density function on $[\![1, n]\!] \times \mathbb{R}_+$ and consider $p = Pf$. Let us express Ωp from definition (3.3) in terms of f:

1. *Degradation.* By integration by parts, we have

$$\forall k \geqslant 1, \quad (k+1)p(k+1) - kp(k) = \int_0^\infty \frac{x^k}{k!} e^{-x} (\partial_x (xf)) dx = P[\partial_x (xf)](k)$$

   whenever boundary terms vanish: a sufficient condition is

$$\lim_{x \to 0} (xf(x)) = \lim_{x \to \infty} (e^{-x} xf(x)) = 0.$$

2. *Creation.* Using the same boundary condition on f, we obtain

$$\forall k \geqslant 1, \quad C[p(k-1) - p(k)] = \int_0^\infty \frac{x^k}{k!} e^{-x} (-C\partial_x f) dx = P[-C\partial_x f](k)$$

3. *Promoter transitions.* Since H does not depend on mRNA, we directly get

$$\forall k \geqslant 0, \quad Hp(k) = H[Pf](k) = P[Hf](k).$$

We obtain (3.8) by gathering the three pieces, forgetting the problem with $k = 0$ in the first two, and recalling that P is injective.

3.B.2 Dual approach

Here we give some details on Proposition 3.4. Recall that by Appendix 3.A, we got well-defined semigroups $S_t = T_t^*$ and $\widetilde{S}_t = \widetilde{T}_t^*$. Thanks to the dual approach, one can derive equations (3.12) and (3.13) by applying T_t to functions

$$f_z : (i, k) \mapsto (\mathbb{1}_{i=1} z^k, \dots, \mathbb{1}_{i=n} z^k)^\top \in c_0^n$$

for $z \in (-1, 1)$, and applying $\widetilde{T}_t$ to functions

$$f_s : (i, x) \mapsto (\mathbb{1}_{i=1} e^{sx}, \dots, \mathbb{1}_{i=n} e^{sx})^\top \in \mathcal{C}_0^n$$

for $s \in (-\infty, 0)$, and using definitions of $\mathcal{L}$ and $\widetilde{\mathcal{L}}$. Let us now derive Proposition 3.4 as a direct consequence. Let $\mu_0 \in \mathcal{M}_n(\mathbb{R}_+)$, $p_0 = P\mu_0$, $p(t) = S_t p_0$ and $\mu(t) = \widetilde{S}_t \mu_0$. Then we

have $G_{p(0)}(z) = L_{\mu(0)}(z-1)$, and it follows that

$$G_{p(t)}(z) = L_{\mu(t)}(z-1)$$

for all $t \geqslant 0$ thanks to (3.12)-(3.13). Thus, by Lemma 3.2, $p(t) = P\mu(t)$ for all $t \geqslant 0$, which can be written $S_t P\mu_0 = P\widetilde{S}_t\mu_0$ or equivalently $P^{-1}S_tP\mu_0 = \widetilde{S}_t\mu_0$.

3.B.3 Path-based approach

Given $(E_t)_{t\geqslant 0}$, the conditional master equation is

$$\partial_t\overline{p}(k,t) = d_0[(k+1)\overline{p}(k+1,t) - k\overline{p}(k,t)] + \overline{u}(E_t)[\overline{p}(k-1,t)\mathbb{1}_{k>0} - \overline{p}(k,t)]$$

where

$$\overline{p}(k,t) = \mathbb{P}(M_t = k|(E_\tau)_{\tau\geqslant 0}) = \mathbb{P}(M_t = k|(E_\tau)_{\tau\in[0,t]})$$

and with $\overline{u}$ as in (3.11). Note that $\overline{p}$ can give back the solution p of the original master equation (3.4) when integrated appropriately.

Appendix 3.C – Spectral properties of promoter transitions

From section 3.1, the transpose of the promoter transition matrix is defined by

$$H_{i,j} = r_{j,i} \quad \text{for } i \neq j \quad \text{and} \quad H_{i,i} = -\sum_{j\neq i} r_{i,j}.$$

In this section, we prove Lemma 3.13 concerning the eigenvalues $a_1^i, \dots, a_N^i$ of "principal submatrices" $-H_{\{i\}} \in \mathbb{R}^{N\times N}$ for $i \in [\![1, n]\!]$ where $n = N+1$. Let us first recall how to derive the fact that 0 is a simple eigenvalue of $-H$ with all its other eigenvalues $b_1, \dots, b_N$ having positive real parts. The main two ingredients are Gershgorin's circle theorem and the Perron-Frobenius theorem.

Let $\alpha = \max_i \left\{ \sum_{j\neq i} r_{i,j} \right\}$. Since H is irreducible, we have $\alpha > 0$ and the matrix

$$M = \frac{1}{\alpha}H + I$$

is irreducible and nonnegative. Its spectral radius $\rho(M)$ satisfies $\rho(M) \geqslant 1$ since 1 is clearly an eigenvalue of M, and it is straightforward to see by Gershgorin's circle theorem and by construction of H that $\rho(M) \leqslant 1$, so $\rho(M) = 1$. Hence 1 is a simple eigenvalue of M by the Perron-Frobenius theorem so 0 is a simple eigenvalue of H. Finally, applying Gershgorin's circle theorem to H shows that the only possible nonnegative eigenvalue of H is 0, so $b_1, \dots, b_N$ all have positive real parts.

Second, consider submatrices $H_{\{i\}} \in \mathbb{R}^{N\times N}$. Once again, Gershgorin's theorem shows that the only possible eigenvalue of $H_{\{i\}}$ with a nonnegative real part would be 0. But by Innocentini *et al.* (2013, Lemma 1), we know that the vector

$$(\det(H_{\{1\}}), \dots, \det(H_{\{n\}}))^\top$$

is a Perron-Frobenius eigenvector of H (i.e. associated with the dominant eigenvalue 0), meaning that all its components are nonzero. As a result, 0 cannot be an eigenvalue of $H_{\{i\}}$

and $a_1^i, \ldots, a_N^i$ all have positive real parts.

Finally, products $\prod_{k=1}^N b_k$ and $\prod_{k=1}^N a_k^i$ for $i \in [\![1, n]\!]$ are real and positive since the related eigenvalues come by conjugate pairs, and standard results on the characteristic polynomial of H show that

$$\prod_{k=1}^{N} b_k = \sum_{i=1}^{n} \prod_{k=1}^{N} a_k^i.$$

Thus, we get the result in terms of the promoter stationary distribution:

$$\mathbb{P}_E(i) = \prod_{k=1}^{N} \frac{a_k^i}{b_k} > 0, \quad \forall i \in [\![1, n]\!].$$

Chapitre 4

Variabilité au cours de la différenciation des cellules

Dans ce chapitre, on laisse la modélisation mathématique de coté et l'on s'intéresse à la variabilité biologique observée dans les niveaux d'expression des gènes lorsque ceux-ci sont mesurés à l'échelle des cellules individuelles. L'article qui suit considère plus spécifiquement la différenciation de progéniteurs érythrocytaires, des cellules en fin de spécialisation et n'ayant plus qu'un seul choix possible : s'auto-renouveler, c'est-à-dire se diviser à l'identique, ou bien se différencier en globules rouges. Les cellules, d'abord maintenues en état d'auto-renouvellement dans un milieu spécifique, sont placées à l'instant $t = 0$ dans un milieu favorable à la différenciation. On mesure ensuite les niveaux d'expression d'une centaine de gènes à $t = 8$, 24, 33, 48 puis enfin 72h où la plupart des cellules sont différenciées. À travers l'évolution de l'expression des gènes, c'est donc la dynamique de la différenciation que l'on peut observer au niveau *single-cell*.

Les données de cellules uniques, bien que prenant la forme de *snapshots* plutôt que de véritables trajectoires des niveaux d'expression dans chaque cellule, ont l'avantage de permettre de retrouver les caractéristiques des données de population tout en contenant une information bien plus riche : la variabilité de chaque gène et les dépendances statistiques entre les gènes, deux caractéristiques qui s'avèrent effectivement présentes. On peut ainsi observer que les corrélations linéaires entre les gènes forment des motifs qui évoluent de manière importante au cours du temps, et l'un des résultats principaux de l'article est de mettre en valeur une augmentation de l'entropie de Shannon des gènes entre 0 et 24h suivie d'une diminution générale. Il y a donc un pic de variabilité autour de 24h, et il se trouve que ce pic semble précéder assez précisément le moment où la plupart des cellules prennent la décision de se différencier. Il se pourrait donc qu'il y ait un lien biologique (i.e. fonctionnel) entre la variabilité des niveaux d'expression et la différenciation des cellules, ce qui est discuté en détail dans l'article.

En outre, un aspect important a consisté à pré-traiter les données d'expression de façon à pouvoir les analyser, dans l'article lui-même mais également dans les chapitres suivants. Il est en effet nécessaire, avant de considérer que les données sont fiables quantitativement, d'étudier les divers problèmes techniques qui peuvent survenir. Ces problèmes ont pu être minimisés par l'utilisation de *spikes*, i.e. des séquences particulières dont l'expression varie en principe extrêmement peu parmi les cellules considérées. Les divers aspects du pré-traitement sont détaillés à la fin de l'article, et dans les chapitres suivants nous utiliserons directement les données sous cette forme.

PLOS | BIOLOGY

RESEARCH ARTICLE

Single-Cell-Based Analysis Highlights a Surge in Cell-to-Cell Molecular Variability Preceding Irreversible Commitment in a Differentiation Process

Angélique Richard[1], Loïs Boullu[2,3,4], Ulysse Herbach[1,2,3], Arnaud Bonnafoux[1,2,5], Valérie Morin[6], Elodie Vallin[1], Anissa Guillemin[1], Nan Papili Gao[7,8], Rudiyanto Gunawan[7,8], Jérémie Cosette[9], Ophélie Arnaud[10], Jean-Jacques Kupiec[11], Thibault Espinasse[3], Sandrine Gonin-Giraud[1☯], Olivier Gandrillon[1,2☯*]

1 Univ Lyon, ENS de Lyon, Univ Claude Bernard, CNRS UMR 5239, INSERM U1210, Laboratory of Biology and Modelling of the Cell, 46 allée d'Italie Site Jacques Monod, F-69007, Lyon, France, **2** Inria Team Dracula, Inria Center Grenoble Rhône-Alpes, France, **3** Université de Lyon, Université Lyon 1, CNRS UMR 5208, Institut Camille Jordan 43 blvd du 11 novembre 1918, F-69622 Villeurbanne-Cedex, France, **4** Département de Mathématiques et de statistiques de l'Université de Montréal, Pavillon André-Aisenstadt, 2920, chemin de la Tour, Montréal (Québec) H3T 1J4 Canada, **5** The CoSMo company. 5 passage du Vercors – 69007 LYON – France, **6** Univ Lyon, Univ Claude Bernard, CNRS UMR 5310 - INSERM U1217, Institut NeuroMyoGène, F-69622 Villeurbanne-Cedex, France, **7** Institute for Chemical and Bioengineering, ETH Zurich, Zurich, Switzerland, **8** Swiss Institute of Bioinformatics, Quartier Sorge - Batiment Genopode, 1015 Lausanne Switzerland, **9** Genethon – Institut National de la Santé et de la Recherche Médicale – INSERM, Université d'Evry-Val-d'Essone – 1 rue de l'internationale 91000 Evry, France, **10** RIKEN - Center for Life Science Technologies (Division of Genomic Technologies)—CLST (DGT), 1-7-22 Suehiro-cho, Tsurumi-ku, Yokohama, Kanagawa 230-0045, Japan, **11** INSERM, Centre Cavaillès, Ecole Normale Supérieure, F-75005 Paris, France

☯ These authors contributed equally to this work.
* Olivier.Gandrillon@ens-lyon.fr

OPEN ACCESS

Citation: Richard A, Boullu L, Herbach U, Bonnafoux A, Morin V, Vallin E, et al. (2016) Single-Cell-Based Analysis Highlights a Surge in Cell-to-Cell Molecular Variability Preceding Irreversible Commitment in a Differentiation Process. PLoS Biol 14(12): e1002585. doi:10.1371/journal.pbio.1002585

Academic Editor: Sarah A. Teichmann, EMBL-European Bioinformatics Institute & Wellcome Trust Sanger Institute, UNITED KINGDOM

Received: June 6, 2016

Accepted: September 22, 2016

Published: December 27, 2016

Data Availability Statement: The eight RNA-seq libraries (raw sequences, 4 conditions, 2 libraries per condition: paired-end libraries) have been deposited at SRA and are available at: http://www.ncbi.nlm.nih.gov/sra/SRP076011. The resulting counting table (matrix_2793.txt), the list of the resulting 424 differentially expressed genes (424_differential_genes.txt), the raw Fluidigm data for cell populations (Pop_1361941161.csv and Pop_VM117_AR2_03-03-14.csv), the raw Fluidigm

Abstract

In some recent studies, a view emerged that stochastic dynamics governing the switching of cells from one differentiation state to another could be characterized by a peak in gene expression variability at the point of fate commitment. We have tested this hypothesis at the single-cell level by analyzing primary chicken erythroid progenitors through their differentiation process and measuring the expression of selected genes at six sequential time-points after induction of differentiation. In contrast to population-based expression data, single-cell gene expression data revealed a high cell-to-cell variability, which was masked by averaging. We were able to show that the correlation network was a very dynamical entity and that a subgroup of genes tend to follow the predictions from the dynamical network biomarker (DNB) theory. In addition, we also identified a small group of functionally related genes encoding proteins involved in sterol synthesis that could act as the initial drivers of the differentiation. In order to assess quantitatively the cell-to-cell variability in gene expression and its evolution in time, we used Shannon entropy as a measure of the heterogeneity. Entropy values showed a significant increase in the first 8 h of the differentiation process, reaching a peak between 8 and 24 h, before decreasing to significantly lower values. Moreover, we observed that the previous point of maximum entropy

data for single cells (0 to 8 hours kinetics: Single _AR78_1_to_2.csv; 0 to 72 kinetics: Single _AR85_1_to_6.csv) the actinomycin D experiment (export_RNA_deg_exp_Diff_0h.csv; export_RNA_deg_exp_Diff_24h.csv; export_RNA_deg_exp_Diff_72h.csv), as well as data for Figures are available at osf.io/k2q5b (DOI 10.17605/OSF.IO/K2Q5B).

Funding: This work was supported by funding from the Institut Rhônalpin des Systèmes Complexes (IXXI), La Ligue contre le Cancer (comité de Haute-Savoie) and by two grants from the French agency ANR (Stochagene; ANR 2011 BSV6 014 01 and ICEBERG; ANR-IABI-3096). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

Abbreviations: CV, coefficent of variation; DNB, dynamical network biomarker; HCA, hierarchical cluster analysis; LDHA, Lactate deshydrogenase A; OSC, oxydosqualene cyclase; PC1, first principal component; PCA, principal component analysis; RT-qPCR, reverse transcription quantitative PCR; SNE, Stochastic Neighbor Embedding.

precedes two paramount key points: an irreversible commitment to differentiation between 24 and 48 h followed by a significant increase in cell size variability at 48 h. In conclusion, when analyzed at the single cell level, the differentiation process looks very different from its classical population average view. New observables (like entropy) can be computed, the behavior of which is fully compatible with the idea that differentiation is not a "simple" program that all cells execute identically but results from the dynamical behavior of the underlying molecular network.

Author Summary

The differentiation process has classically been seen as a stereotyped program leading from one progenitor toward a functional cell. This vision was based upon cell population-based analyses averaged over millions of cells. However, new methods have recently emerged that allow interrogation of the molecular content at the single-cell level, challenging this view with a new model suggesting that cell-to-cell gene expression stochasticity could play a key role in differentiation. We took advantage of a physiologically relevant avian cellular model to analyze the expression level of 92 genes in individual cells collected at several time-points during differentiation. We first observed that the process analyzed at the single-cell level is very different and much less well ordered than the population-based average view. Furthermore, we showed that cell-to-cell variability in gene expression peaks transiently before strongly decreasing. This rise in variability precedes two key events: an irreversible commitment to differentiation, followed by a significant increase in cell size variability. Altogether, our results support the idea that differentiation is not a "simple" series of well-ordered molecular events executed identically by all cells in a population but likely results from dynamical behavior of the underlying molecular network.

Introduction

The classical view of a linear differentiation process driven by the sequential activation of master regulators [1] has been increasingly challenged in the last few years both by experimental findings and theoretical considerations.

Thanks to the recent development in single-cell profiling technologies, researchers are now able to investigate qualitatively and quantitatively the cell-to-cell variability in gene expression in more detail. In this context, several experimental studies at single-cell level involving the regulation of self-renewal and differentiation processes in embryonic stem cells [2–8] and the generation of induced pluripotent stem cells [9] have shown that gene expression variability might be involved in cell differentiation. To support this claim, recent researches on hematopoietic stem cells highlighted the role of molecular heterogeneity in differentiation [10, 11]. Further evidence was also obtained during an ex vivo differentiation process [12], and in the generation of cells of the immune system [13–18].

The overt cell-to-cell variability is deeply rooted in the inherent stochasticity of the gene expression process [19–23]. Numerous explanations have been put forward regarding the molecular and cellular sources for such variability (see [24] and references therein). Some of those causes involve biophysical processes (e.g., the random partitioning during mitosis, as

discussed in [25]), whereas others are more related to biochemical regulation (e.g., the dynamical functioning of the intracellular network [26] or the chromatin dynamics [27]).

At least three models of cell differentiation based on stochastic gene expression have been proposed, in which a peak in the gene expression variability is expected to occur. In the first model, stochastic gene expression is the driving force of cell differentiation that generates cell type diversity, on which a selective constraint is then exerted [28]. In the second model, noise in gene expression causes bifurcations in the dynamics of gene regulatory networks [21]. In the third model, cell differentiation is viewed as a dynamical process in which differentiating cells are thought of as particles moving around in a state space [29, 30]. This formal space can be used to display gene expression patterns. Hence, when some parameters that describe gene regulatory interactions change, the cell particle "moves" in the state space. In this view, discrete identified cell states (e.g., self-renewing, differentiated) correspond to different regions of this space that could be seen as different attractor states. The transition process between attractors therefore first requires the exit from the original state that may be fueled by an increase in gene expression stochasticity [31]. Regardless of the differences between these models, they all assume that the differentiation process is represented by cell trajectories leading from one state to another through a phase of biased random walk in gene expression. This phase is followed by stabilization (convergence) toward a particular pattern of gene expression corresponding to a stable attractor state, the differentiated final state, in which noisy fluctuations of gene expression is minimized by the stabilizing effect of the attractor. Therefore, changes in the extent of cell-cell variability could be a new observable metric to characterize the cell differentiation process.

The purpose of the present study was then to assess whether gene expression variability changes during the differentiation process, as suggested by the above-quoted models, and whether such variation concurs with any physiological cellular change. We investigated the extent of gene expression variability at the single-cell level, both before and during the cell differentiation process. To do this, we analyzed the differentiation process of T2EC, which is an original cellular system consisting of non-genetically modified avian erythrocytic progenitor cells grown from a primary culture [32]. These cells can be maintained ex vivo in a self-renewal state under a combination of growth factors (TGF-α, TGF-β, and dexamethasone) and can also be induced to differentiate exclusively toward erythrocytes by changing the combination of the external factors present in the medium. The primary cause for differentiation is therefore known and relies upon change in the information carried by the extracellular environment. The differentiation process in those cells has been previously analyzed at the population level [33–35].

We first selected a pool of 110 relevant genes on the basis of RNA-Seq analysis performed on populations of T2EC in self-renewal state or induced to differentiate for 48 h. Multivariate statistical analysis of the data allowed us to select 92 genes for further analysis. We then performed high-throughput reverse transcription followed by reverse transcription quantitative PCR (RT-qPCR) of the 92 selected genes on single-cells collected at six time-points of differentiation. Several dimensionality reduction algorithms were used to visualize trends in the datasets. In agreement with the above hypothesis, cell heterogeneity, as measured by entropy, significantly increased during the first hours of the differentiation process and reached a maximal value at 8 to 24 h before decreasing toward the end of the process. The peak in entropy preceded an increase in cell size variability at 48 h. These observations suggested that 24 h is a crucial turning point in the erythrocytic differentiation process, which was experimentally verified by showing that T2EC committed irreversibly to the differentiation process between 24 h and 48 h.

Results

Identification of Differentially Expressed Genes Between Self-Renewing and Differentiating Progenitors

In order to identify a pool of genes potentially relevant in the differentiation process, we analyzed the transcriptome of self-renewing and differentiating primary chicken erythrocytic progenitor cells (T2EC) using RNA-Seq. We sequenced two independent libraries from self-renewing T2EC and two independent libraries from T2EC induced to differentiate for 48 h. For each condition, we first verified that read counts between replicates were reproducible (S3A and S3B Fig). We then identified 424 significantly differentially expressed genes (p-value < 0.05, S3C Fig). Gene ontology analysis using the DAVID database [60] revealed a clear over-representation of genes involved in sterol biosynthesis in this list (not shown). This finding was in line with our previous analysis showing that the oxydosqualene cyclase (OSC), which is involved in cholesterol synthesis, is required to maintain self-renewal in T2EC [35]. However, no other over-represented function emerged from the present analysis.

Identification of Genes Relevant to Analyze the Erythrocytic Differentiation Process

To identify a smaller subset of relevant genes for further analysis by RT-qPCR using the Fluidigm array (see below), we tested 56 down-regulated and 77 up-regulated genes among the above 424 genes differentially expressed in self-renewing versus differentiating cells, which had the smallest set of p-values. We also included 32 non-regulated genes, selected among the most invariant ones. We then measured the expression of these 165 genes first using RNA from bulk cell populations taken at five time-points during differentiation (0, 8, 24, 48, and 72 h). Based on qPCR primer efficiency, 55 genes were removed (see Materials and Methods), which left a total of 110 genes for the subsequent analysis.

A principal component analysis (PCA) on the bulk gene expression levels (Fig 1A) showed a clear separation of the time-point 0 h (self-renewal) from the differentiation time-points. Samples along the differentiation process were well ordered according to the first principal component (PC1). PC1 explained 56.2% of the data variability suggesting that the differentiation process is the main source of variability at the population level for the selected genes.

We also performed a hierarchical cluster analysis (HCA), which again showed a clear arrangement of the samples according to their position along the differentiation process (Fig 1B). We further noticed that the gene expression patterns at 0, 8, and 24 h time-points were more similar to each other, while those at 48 h and 72 h time-points were also more similar to each other.

Thus, the 110 selected genes allowed us to clearly distinguish cell populations according to their progression along the differentiation sequence, indicating that they were relevant for analyzing this process. However, since the single-cell measurement technology used in this study could only accommodate 92 genes (not including two spikes and two repeats for the *RPL22L1* gene), we further refined our gene choice by performing a K-means clustering on the above data. The algorithm grouped genes based on their expression profile, and identified seven different gene clusters with respect to expression kinetics (S4 Fig).

The patterns mainly showed decreasing or increasing gene expressions during the differentiation process, while one cluster displayed a more complex dynamic (cluster 4). The latter was composed of genes whose expression decreased during the first 8 h, then increased and stabilized between 24 h and 48 h, before decreasing again until 72 h. Interestingly, all genes belonging to this cluster were linked by their involvement in sterol biosynthesis, reinforcing the

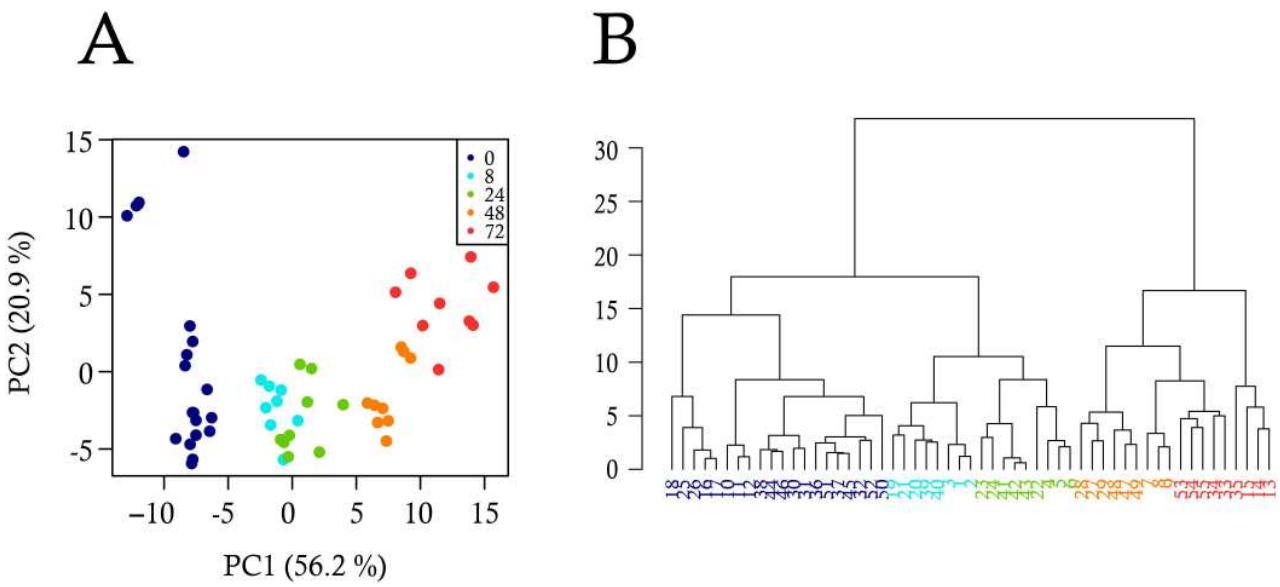


Fig 1. Analysis of bulk-cell gene expression during the differentiation process. Gene expression data were produced by RT-qPCR in triplicate from three independent T2EC populations collected at five differentiation time-points (0 h, 8 h, 24 h, 48 h, 72 h). The expression level of 110 genes (18 invariants, 50 down-regulated and 42 up-regulated) was analyzed by two different multivariate statistical methods: (A) Principal component analysis (PCA), and (B) Dendogram resulting from hierarchical cluster analysis (HCA). The dots in (A) and leaves in (B) indicate the different cell populations and the colors indicate the differentiation time-points at which they were collected.

doi:10.1371/journal.pbio.1002585.g001

previously noted role of this pathway in erythroid differentiation. Based on the result of K-means clustering, we selected around thirteen genes per group to represent each cluster equally. This left us with 92 genes for further analysis (S1 Table).

We then used STRING database to search for known connections among these genes. The result confirmed the existence of a strongly connected subnetwork associated with sterol synthesis (S5B Fig). Moreover, this analysis also revealed the presence of another highly connected subnetwork mostly composed of genes involved in signaling cascades and two transcription factors (BATF and RUNX2). Those two main networks are linked by the gene *HSP90AA1* which encodes the molecular chaperone HSP90*alpha*. Its activity is not only involved in stress response but also in many different molecular and biological processes because of its important interactome. HSP90*alpha* represents 1%–2% of total cellular protein in unstressed cells. Interestingly, HSP90*alpha* level is up-regulated and correlated with poor disease prognosis in leukemia [61]. HSP90*alpha* has also been shown to be involved in the survival of cancer cells in hypoxic conditions [62].

Cell-to-Cell Heterogeneity Blurred Cell Differentiation Process

We measured the expression level of the selected 92 genes by single-cell RT-qPCR using 96 cells isolated from the most informative time-points of the differentiation sequence. Based upon preliminary experiments, we decided to analyze cells from six time-points during differentiation. After data cleaning (see Materials and Methods), we obtained the expression level of 90 genes in 55, 73, 72, 70, 68, and 51 single cells from 0, 8, 24, 33, 48, and 72 h of differentiation, respectively.

One should note that the variability we observed at the single-cell level originates from two types of sources: biological sources and experimental sources. We therefore tested the

technical reproducibility of different RT-qPCR steps liable to generate such experimental noise (see Materials and Methods). As expected, reverse transcription (RT) was the main source of experimental variability, since pre-amplification and qPCR steps brought negligible amount of variability (S1 Fig). Moreover, using external RNA spikes controls whose Cq value depends only on the experimental procedure, we noted that technical variability was negligible compared to the biological variability (see Materials and Methods). Quality control (see Materials and Methods) led to the elimination of 2 genes, letting us with 90 genes for subsequent analysis.

We first used PCA on the single-cell expression of these 90 genes (Fig 2A). In contrast to the whole-population data, the single-cell data did not immediately demarcate into well-separated clusters. The differentiation process was most apparent by looking at the second principal component (PC2), which explained 9.9% of the variability in the dataset. Hence, unlike in the population-averaged data, the differentiation process did not represent the main source of variability at the single-cell level.

The application of HCA further confirmed that the classification became more complex for single-cell data (Fig 2B). Contrary to bulk analysis, individual cells from the same time-point

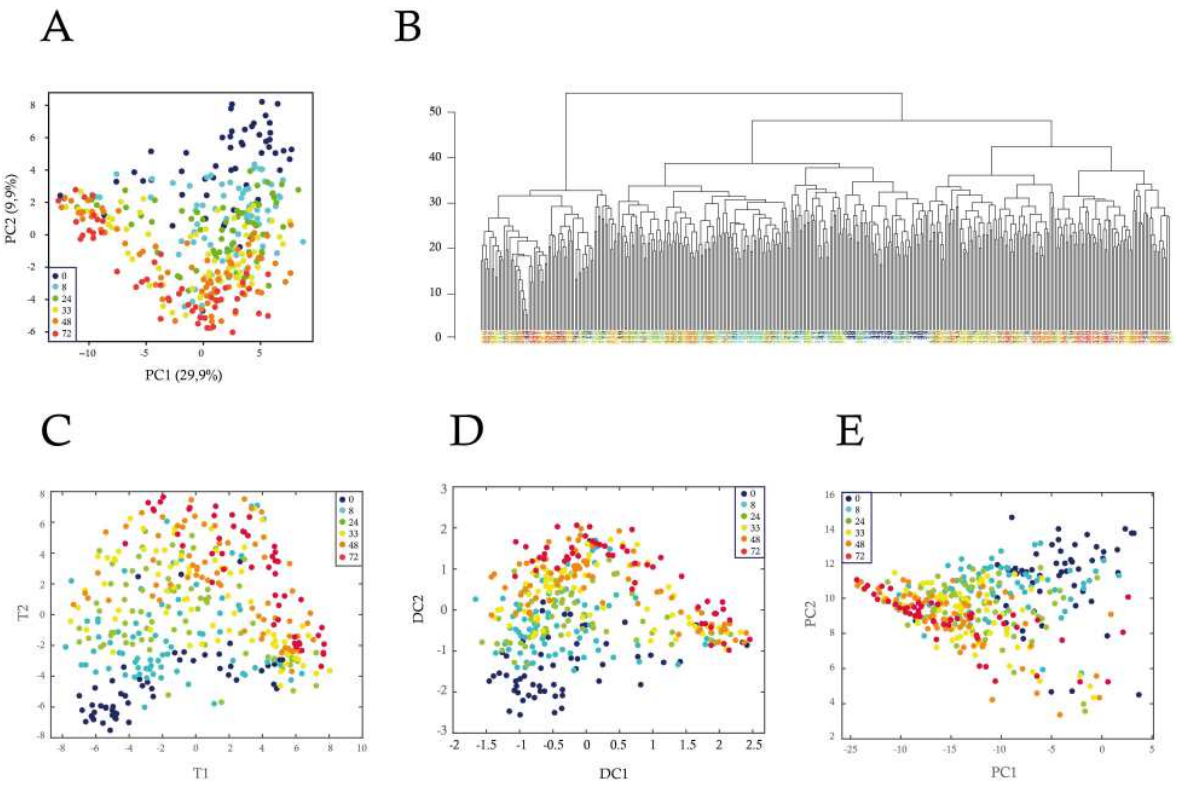


Fig 2. Analysis of single-cell gene expression during the differentiation process. Gene expression data were produced by RT-qPCR from individual T2EC collected at six differentiation time-points (0, 8, 24, 33, 48, and 72 h). The expression of 90 genes was analyzed in single-cells by five different multivariate statistical methods: (A) Principal component analysis (PCA), (B) Hierarchical cluster analysis (HCA), (C) t-SNE, (D) Diffusion map, and (E) kernel PCA. The dots in (A, C, D, and E) and leaves in (B) indicate the single-cells, and the colors indicate the differentiation time-points at which they were collected. t-SNE analysis was performed using the following parameters: initial_dims = 30; perplexity = 60. Diffusion map was run using the following parameters: no_dims = 4, t = 1, and sigma = 1000. Kernel PCA was run with a parameter for computing the "poly" and "gaussian" kernel of 0.1. Only the first two dimensions are plotted.

doi:10.1371/journal.pbio.1002585.g002

were not necessarily more similar to each other than to cells from neighboring time-points. Consequently, the clustering of individual cells into groups became complicated. The picture of cell differentiation process that emerged from the single-cell analysis thus far was more complex than the one obtained from the population level analysis. This difference between single-cell and population-level analysis arises from the unraveling of cell-to-cell heterogeneity in the single-cell data, which could have been hidden by the averaging effect of the population (see below).

PCA is a linear method for dimensionality reduction of single-cell data. In view of non-linear relationships of cell states in state space, recently nonlinear techniques like t-SNE [55] or diffusion maps [63] have been applied in single-cell data analysis. t-SNE is a variation of Stochastic Neighbor Embedding deemed capable of capturing more local structures than classical PCA, while also revealing global structure such as the presence of clusters at several scales. Diffusion maps use a non-linear distance metric (referred to as diffusion distance), which is deemed conceptually relevant in view of noisy diffusion-like dynamics during differentiation [63]. We therefore applied these algorithms on our datasets, as well as another non-linear version of PCA, called Kernel PCA [64], not previously applied to single-cell gene expression data (Fig 2C to 2E). The general conclusions obtained by PCA did not appreciably change when using these non-linear dimensionality reduction techniques. There was again an obvious trend reflecting the differentiation process, as well as a significant amount of intermingling of cells from different time-points.

Single-Cell Data Embed Population Information and Reveal New Discriminating Genes Involved in the Differentiation Process

In order to assess to what extent the differentiation process was still visible in the single-cell data, we performed PCA on datasets from the two extreme time-points, 0 and 72 h (Fig 3A). The result showed a clear separation of both time-points with only a few cells intermingled. We also performed HCA on datasets from the same time-points (Fig 3B). Again, the segregation of the cells was still not perfect, but cells were not as mixed as before. Here, there exist two clusters of self-renewing and differentiating cells. When compared to the analysis of the entire time series, the separation between cells from the two extreme time-points looked clearer. Therefore, the analysis of single-cell data confirmed that part of the information present in the single-cell data is linked to the differentiation process.

The idea that shared information was present in single-cell and population-based data was reinforced by the analysis of the correlation matrices within and between the two datasets (S6 Fig). It was apparent that (1) the global intensity of the correlations was higher with population-based data and (2) there existed a co-structure between the two datasets. At the population level, we showed that the set of genes selected was relevant to analyze the differentiation process (Fig 1). The cross-correlation analysis strengthened this view and demonstrated that when looking at the single-cell scale, the information held by these genes was not totally erased by cell-to-cell variability.

We then looked at the genes that contributed the most to the PCA outcome (Fig 3C). Among the genes that discriminate the most self-renewing cells, one could highlight *LDHA* (Lactate deshydrogenase A), *CRIP2*, and *Sca2*. *Sca2* is a gene that we previously have shown to be associated with the self-renewal of erythroid progenitors [34]. *LDHA* is less expected and will be discussed below. Among the genes that contributed the most to discriminating differentiated cells, one could highlight *RHPN2* and *betaglobin*. Since betaglobin is a part of hemoglobin, the most abundant protein in erythrocytes, it was expected to be associated with differentiating cells.

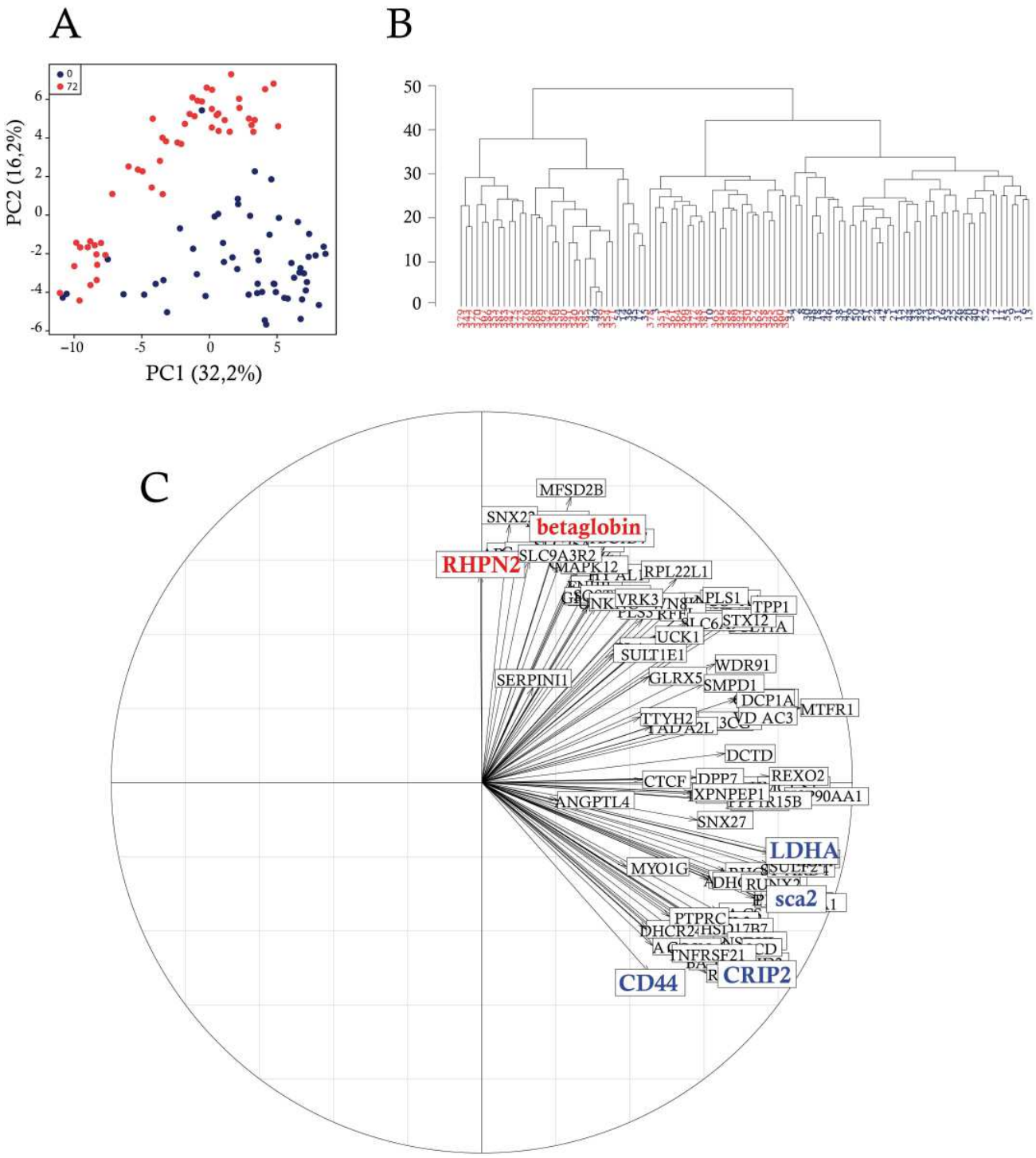


Fig 3. Gene expression-based discrimination between self-renewing and differentiating individual cells. Single-cell gene expression data were analyzed considering only self-renewing cells and cells induced to differentiate since 72 h. (A) Principal component analysis (PCA); (B) Hierarchical cluster analysis (HCA) was used to sort single-cells picked up at 0 h and 72 h of the differentiation process according to similarity measurement; (C) Two-dimensional representation of the contribution of each variable (gene) to the inertia. The direction of the arrows displays the contribution of that variable to the underlying component. The colored genes highlight genes of interest and genes that contributed the most to the PCA outcome, associated with self-renewal (blue) and the erythroid differentiation process (red).

doi:10.1371/journal.pbio.1002585.g003

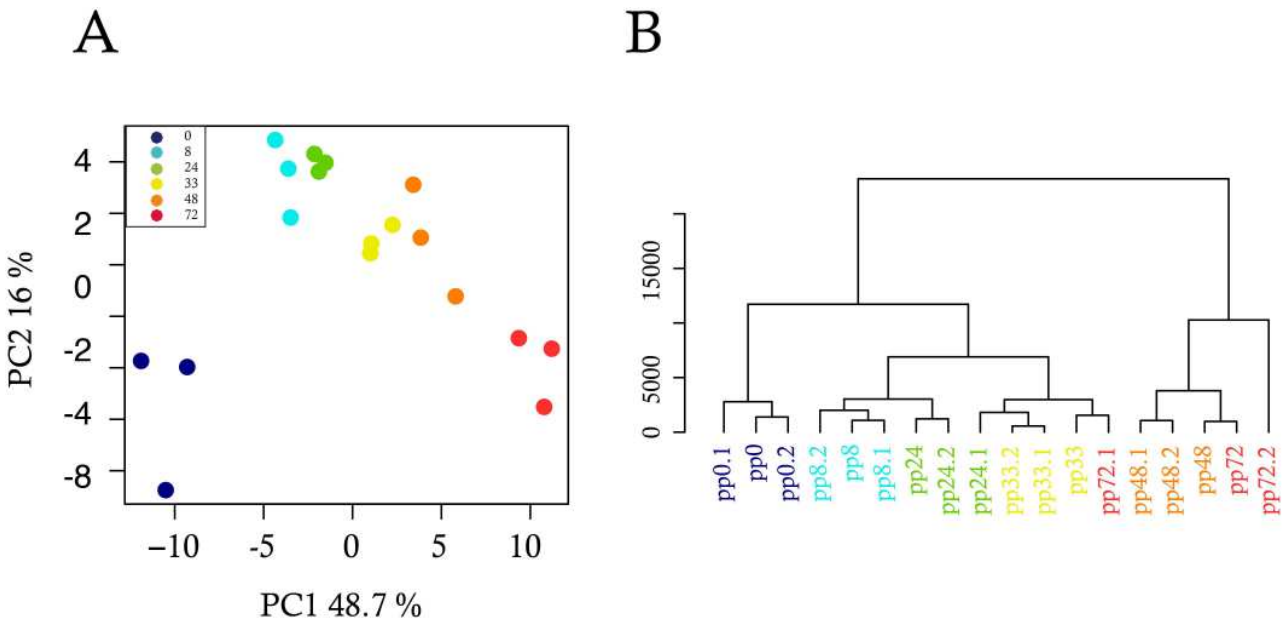


Fig 4. Analysis of single-cell data averaged over pseudo-populations. We separated single-cells into three pseudo-populations with around one-third of single cells for each time-point. We then calculated the average gene expression over each pseudo-population, and analyzed the resulting averaged data using multivariate statistical methods. (A) Principal component analysis (PCA); (B) Hierarchical cluster analysis (HCA).

doi:10.1371/journal.pbio.1002585.g004

Single-Cell Data Averaging Recapitulates Results from Population-Level Analysis

Given that the analysis of single-cell gene expression did not produce a clear separation of the temporal stages, in contrast to whole populations, we hypothesized that by averaging over a population of individual cells, we should be able to reproduce the bulk results. For this purpose, we generated three pseudo-populations (sub-populations) of about one-third of cells from the single-cell data and computed their average gene expressions for each time-point. By performing PCA on the mean gene expressions of these pseudo-populations, we noticed that the averaged data showed more organization and, importantly, that the differentiation progression materialized along the PC1 dimension (Fig 4A).

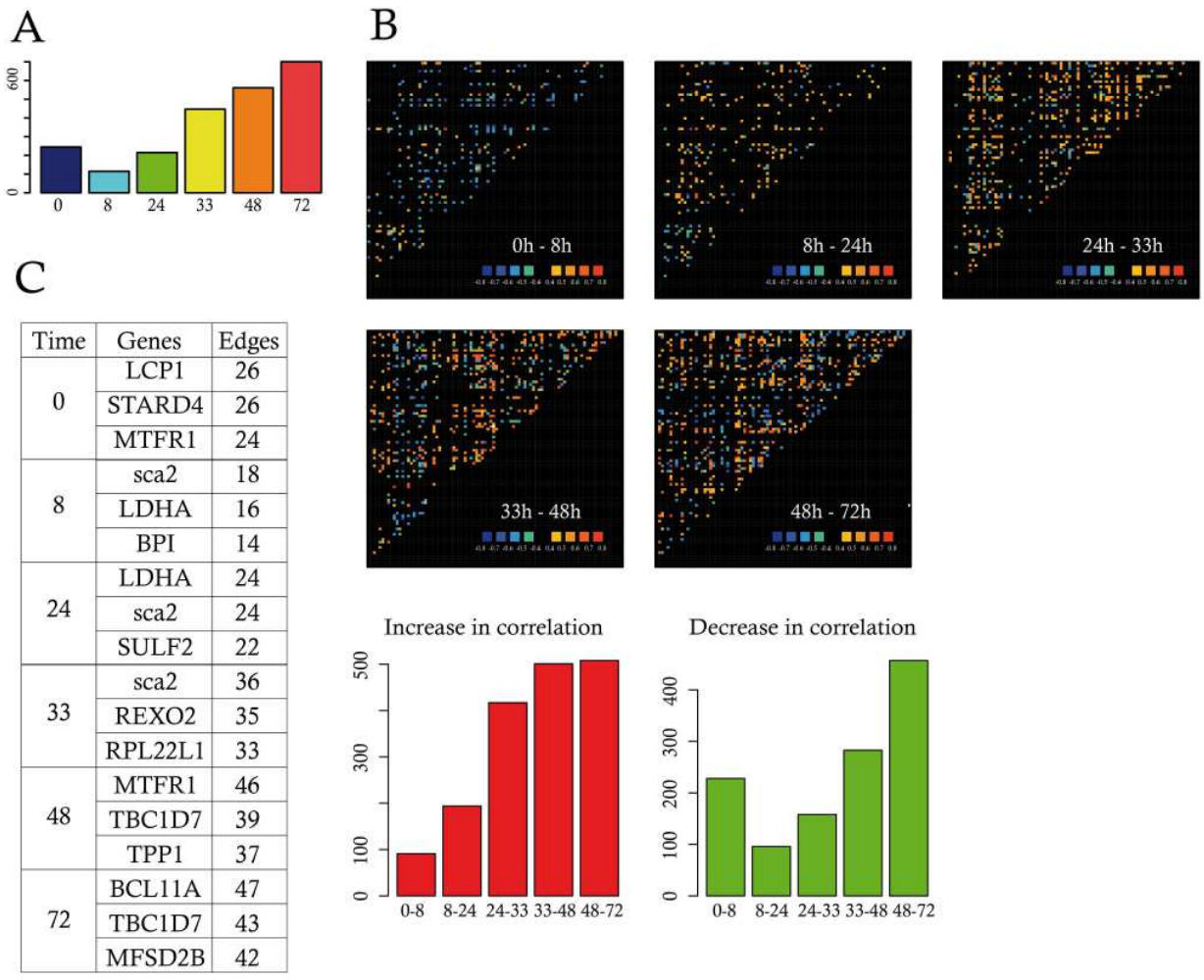
The PCA result of the pseudo-population therefore looked much more like the population than the single-cell results. Similarly, HCA generated a clustering that was not quite as clear as the analysis of bulk RNA data, but much better than the single-cell analysis (Fig 4B). The HCA results showed for example similarities between gene expressions from time-points 48 and 72 h. Together the pseudo-population analysis obtained by statistical averaging of single-cell data mostly recapitulated, albeit not entirely, the population-based results, suggesting that the clear-cut classification of bulk-cell-based data is due to the (physical) averaging effect in populations, in line with a previous account [65].

The Correlation Networks are Very Dynamical Entities

Single-cell data offers access to the patterns of the relationship of genes with respect to both their marginal (S7 Fig), as well as their full joint distribution (not shown). This provides us with a new observable that we used to characterize the progression of the differentiation process in finer details.

For each time-point, we computed a correlation matrix to evaluate how correlated the expression of any pair of genes was, across all cells at a given time. Since data were log-normally distributed, we employed the Spearman correlation coefficient. We then calculated the significance of the correlation and used a *p*-value below 0.05 as a cutoff. Two genes (the nodes of a graph) that exhibited a significant correlation were connected by an edge. Finally, we subsampled 85% of the cells for 10,000 iterations, so as to obtain robust correlation networks that will not depend upon the sampling process. We then constructed a gene correlation network for each time-point. Although both positive and negative correlations were computed, negative correlations proved much less robust and were eliminated by the sub-sampling process, in which we only kept significant correlations that appeared in all of the 10,000 subsampling.

As shown in (Fig 5A), the density of the resulting networks (number of significant correlations) was clearly varying along the differentiation process.



C

Time	Genes	Edges
0	LCP1	26
	STARD4	26
	MTFR1	24
8	sca2	18
	LDHA	16
	BPI	14
24	LDHA	24
	sca2	24
	SULF2	22
33	sca2	36
	REXO2	35
	RPL22L1	33
48	MTFR1	46
	TBC1D7	39
	TPP1	37
72	BCL11A	47
	TBC1D7	43
	MFSD2B	42

Fig 5. Gene expression correlations. (A) Shown is the number of significant correlations, between any pair of genes, surviving 10,000 subsampling iterations, per time-point; (B) Correlation variations between two consecutive time-points using the color code bar shown at the bottom right of the panels. Cold colors (blue and green) indicate decreasing genes correlations and hot colors (from yellow to red) stand for increasing gene correlations between the time-points considered. Intermediary variations (between −0.4 and +0.4) as displayed in black. The bottom left red barplot indicates the number of increasing correlations, whereas the green barplot shows the number of decreasing correlations between each pair of consecutive time-points; (C) The three genes that displayed the highest number of edges at each time-point were listed in the table, as well as the number of edges connecting those genes. Data for this figure (A and B) can be found at osf.io/k2q5b.

doi:10.1371/journal.pbio.1002585.g005

One observed a sudden drop in the number of correlations by 8 h that then steadily increased to reach a maximum value at 72 h much higher than the initial value. Interestingly, this global behavior resulted from both an increase and a decrease in gene-to-gene correlation values (Fig 5B). Even between 48 and 72 h, some gene pair correlation decreased while the overall net balance resulted in a global increase.

This fast-changing density of the networks was also accompanied by a progressive change in the identity of the most highly correlated nodes (Fig 5C). Both *Sca2* and *LDHA* that were previously identified by the PCA also appeared as prominent among the correlation network from 8 to 24 h, while later time-points were characterized by the appearance of other genes as *TBC1D7* and *BCL11A*.

One should note that such correlation networks are to be seen as resulting from the behavior of the underlying mechanistic gene interaction networks, but can not be taken per se as a faithful representation of such dynamical interaction networks.

Evidence for the DNB Theory

Contrary to previous accounts [12, 66], we observed a global decrease in the correlation intensity between 0 and 8 h. Nevertheless, we noticed that some gene pairs showed an increased correlation coefficient. We therefore reasoned that those genes could represent a putative dynamical network biomarker (DNB), a subgroup of genes involved in the critical transition phase of a dynamical system [51]. To qualify for a DNB, three conditions have to be fulfilled: (1) the coefficent of variation (CV) of each variable in the DNB should increase, (2) the correlation (PCCin) within the DNB should increase, and (3) the correlation (PCCout) between the DNB and outside genes should decrease. All three conditions can be simultaneously quantified using the I score (see Materials and Methods). We therefore first selected a group of 12 genes by a two-stage process: (1) we first selected all of the genes that participated in at least one pair that showed an increased correlation of at least 0.5 between 0 and 8 h and (2) among those genes, we selected the genes that showed an increase in their CV value between 0 and 8 h. We then computed the I score of that group of genes at each time-point (Fig 6).

Although PCCin slightly decreased with time, this group of genes nevertheless might still qualify for a DNB since they matched two out of the three criteria used to identify DNBs. Their I value first sharply increased before returning to lower values. This rise is mostly due to a sharp decrease in PCCout between 0 and 8 h, accompanied by a more modest increase in CV. As mentioned, the internal correlation value PCCin decreased, and therefore was not driving the I value. One must note that we computed a Pearson correlation coefficient as advocated [51]. We also tried a Spearman correlation value, which showed a slightly different behavior with a modest increase in PCCin between 8 and 24 h and continued to increase steadily up to 72 h, not affecting the global surge in I value (not shown).

The Initial Driver Genes belong to the Sterol Synthesis Pathway

Since we observed major changes after 8 h of differentiation, one asked how early changes in gene expression could be detected. For this we performed a second single-cell kinetic experiment, where we obtained the expression level of 90 genes in 48, 48, 39, and 41 single cells from 0, 2, 4, and 8 h of differentiation, respectively.

We then defined the first wave of response as genes that showed a significant difference between 0 and 2 h. Two genes satisfied this criterion (Fig 7), establishing that the transcriptional response to the medium change was a very fast process, but concerned only a very limited number of genes.

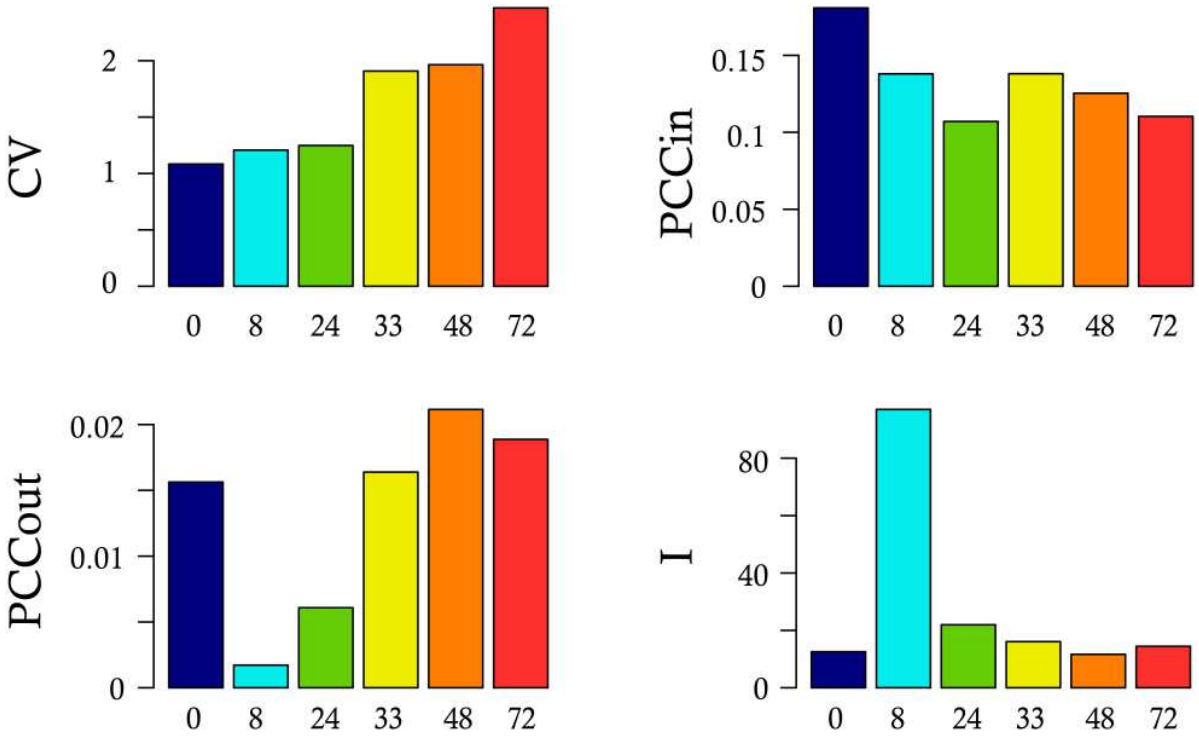


Fig 6. Identification of a dynamical network biomarker. Shown is the behavior of a subset composed of 12 genes fitting the following criteria: increase in their standard deviation and participation to increasing correlations, between 0h and 8h. For this subset, we plotted the mean coefficent of variation (CV), the mean of the correlation between any pair of genes belonging to the subset (PCCin), the mean of the correlation between any one gene of the subset and any one gene outside of the subset (PCCout) and the resulting I-scores, at each time-point. The DNB group included the following genes: *ACSS1*, *ALAS1*, *BATF*, *BPI*, *CD151*, *CRIP2*, *DCP1A*, *EMB*, *FHL3*, *HSP90AA1*, *LCP1*, *MTFR1*. Data for this figure can be found at osf.io/k2q5b.

doi:10.1371/journal.pbio.1002585.g006

The second wave was defined as genes not belonging to wave 1 and showing a significant difference between 2 and 4 h of the response. Five genes satisfied this criterion (Fig 7). It was remarkable that six out of the seven genes from waves 1 and 2 belonged to the same functional group, that is the group of genes associated with sterol synthesis. This proved to be highly statistically significant ($p = 1.8 \times 10^{-6}$). We therefore can propose that the sterol synthesis pathway could act as one of the drivers of the changes that will update the internal network from the changes in external conditions. This would be in line with our previous demonstration for the role of cholesterol synthesis in the decision making process in our cells [35].

A Surge in Cell-to-Cell Variability

A critical novel opportunity provided by single-cell analysis is to study cell-to-cell variability of gene expression as an observable per se and also to add new insight to characterize the temporal progression of differentiation. The question as to what may be the best metrics for quantifying gene expression variability is still open. An aggregated measure called the Jensen-Shannon divergence has been proposed previously as a measure for gene expression noise [9]. One of the main drawbacks of this metric is that it was not possible to assess whether or not the differences observed were statistically significant. We therefore decided to use a simpler Shannon measure of the heterogeneity among the cells for their gene expression profile (see Materials and Methods and S2 Fig). Such a measure provided a distribution of entropy values per gene

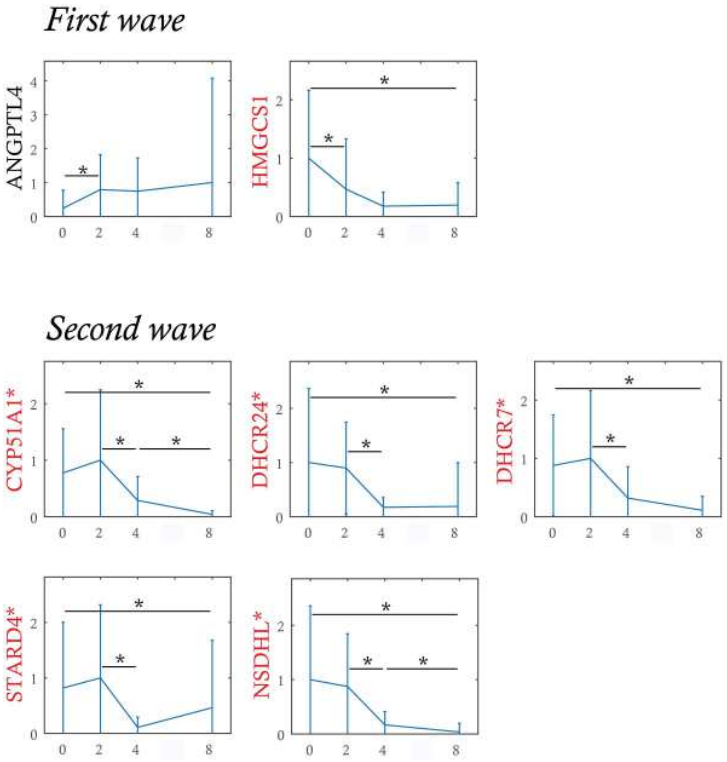


Fig 7. Initial expression waves analysis. Genes are sorted according to the time of the first significant expression variation. The first wave corresponds to genes with a significant variation detected during 0 h and 2 h. The second wave corresponds to genes with a significant variation detected during 2 h and 4 h but without significant variation detected earlier. Genes labeled in red belong to the group of genes associated with sterol synthesis. Significant variations (-*-) are detected by non-parametric Mann-Whitney test (p-value < 0.05) if the test is positive in more than 90% of 1,000 bootstrap samples. Genes prefixed by * have a significant variation between 0 h and 8 h detected in both experiments (0 to 72 h, as well as 0 to 8 h). The probability of having 6 genes over 7 (in the first and second waves) belonging to the 10 sterol cluster genes among all 90 genes is estimated to $p = 1.8 \times 10^{-6}$ with the hypergeometric probability density function. Data for this figure can be found at osf.io/k2q5b.

doi:10.1371/journal.pbio.1002585.g007

per time-point, allowing to perform statistical tests. We observed that this entropy increased gradually along the differentiation process, reaching its maximal value at 8 to 24 h, before declining toward 72 h (Fig 8A).

Such an increase of entropy between 0 and 8h resulted from a global increase of each gene entropy, except for a few (Fig 8B). The observed rise in entropy value was highly significant as early as 8 h when compared to 0 h of differentiation. Furthermore, decrease in entropy also became significant between 24 and 33 h of differentiation (Fig 8C). Consequently, since entropy can be defined as a measure of the disorder of a system, this result suggested that a maximal heterogeneity was achieved at 8–24 h of the differentiation process in the expression of our 90 genes, before significantly decreasing to a much lower level of heterogeneity.

Potential Explanation for the Rise in Variability

Different potential causes can be envisioned to explain this increase in entropy, including cell size and cell-cycle stage variations, asynchrony in the differentiation process, and more dynamical causes.

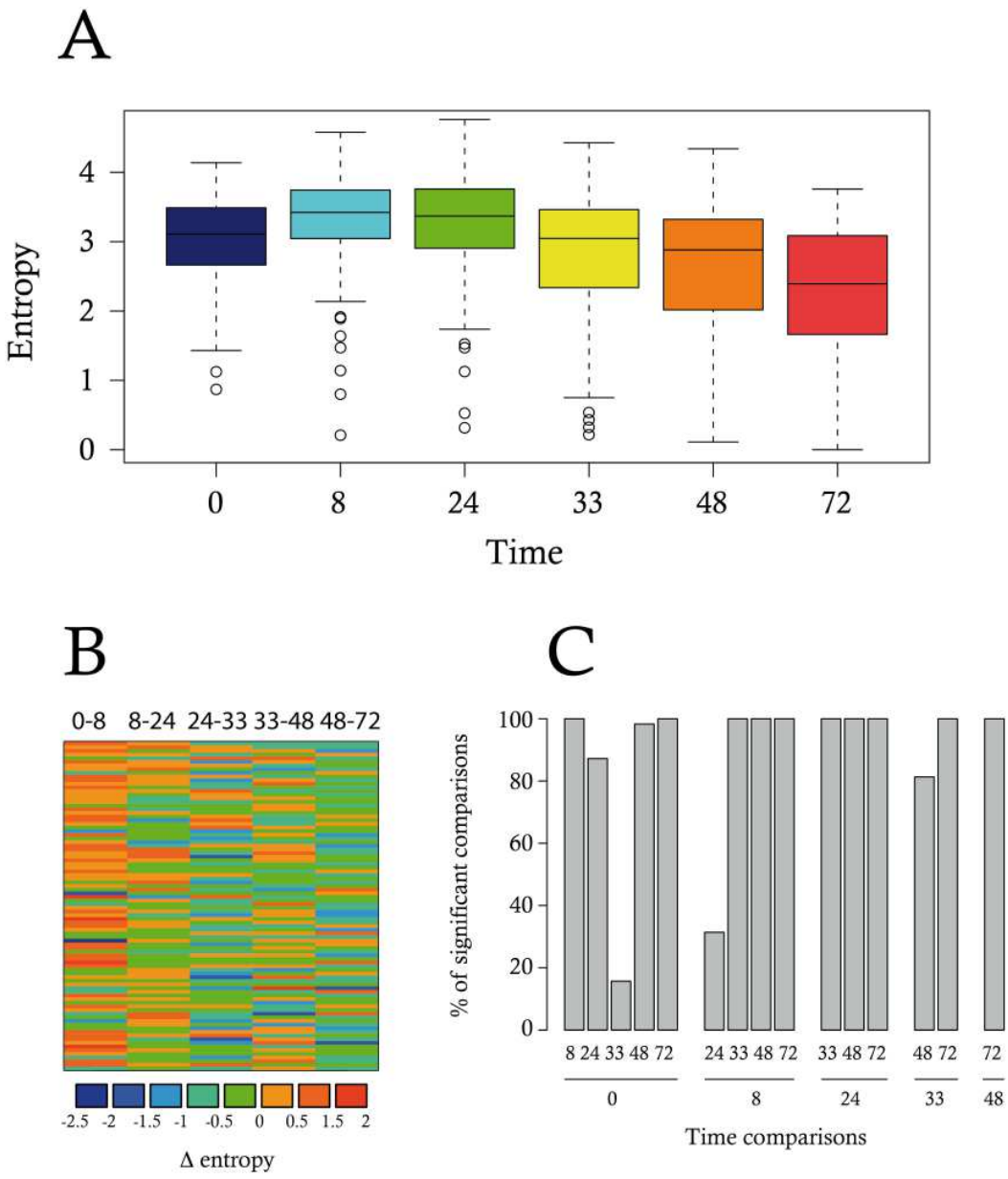


Fig 8. Cell-to-cell heterogeneity measurement using Shannon entropy. (A) A Shannon entropy was calculated for each time-point for each gene. Boxplots represent the distribution of the entropy values; (B) Gene entropy variation: for each gene (i.e., lines), we represented the difference between entropy values at two consecutive time-points (Δ-entropy) using a color gradient code. Negative and null delta entropies (i.e., for a given time-point, the entropy value for these genes decreased or does not change, compared to the earlier time-point) are colored in blue and green. Positive delta entropies are colored in orange or red; (C) We assessed the significance of the differences between any pair of time-point through a Wilcoxon test. The robustness of the result was assessed by performing subsampling. The barplot shows the results as the percentage of 1,000 iterations for which a significant difference (p-value < 0.05) was detected. Data for this figure can be found at osf.io/k2q5b.

doi:10.1371/journal.pbio.1002585.g008

As suggested in some previous works, cell size and cell-cycle stage variations could influence gene expression, and become confounding factors [67–69]. Nevertheless, variability due to variations in cell cycle has been shown to be quantitatively negligible in erythroid precursors [70]. We also added in our gene list the CTCF gene, known to be cell-cycle regulated in chicken cells [71]. Almost no correlation was detected between this gene and any of the 91 other genes (Fig 9A) demonstrating that our gene list contained virtually no other cell-cyle-regulated gene. Furthermore, we assessed whether or not the repartition of our cells within the different phases of the cell cycle could have been modified at a time where entropy was peaking. No significant difference in cell cycle repartition could be seen at 8 h of differentiation (Fig 9B). Altogether, those results demonstrate that a potential effect of cell cycle variation would only marginally explain our data. Regarding cell size, it is important to note that in our system the peak in gene expression variability at 8–24 h occurs at a time where cell size is not affected (Fig 10B). If anything, we observed a slight increase in cell size, which could be responsible for a decrease, and not an increase, in noise [72].

We then assessed a potential effect of asynchrony in the differentiation process. For this, we first employed the following algorithms: SCUBA [52], WANDERLUST [53] and TSCAN [54] to reorder the cells according to the calculated pseudotimes. However, SCUBA led to a cell re-ordering that was highly inconsistent with the actual time-points, where all self-renewing cells (time 0 h) were placed in the middle of the SCUBA order (not shown). WANDERLUST and TSCAN produced a more reasonable cell ordering. However, the trajectories of the gene expression profiles following this ordering were quite erratic (not shown). Nevertheless, the entropy of sub-populations of cells, grouped according to either their WANDERLUST pseudotimes or TSCAN clusters, showed the same rise-then-fall profile as with the original single cell data (Fig 9C and 9D).

In theory, these algorithms are supposed to reconstruct a posteriori the "hidden" order along the differentiation pathway. Within this frame, the behavior of entropy in re-ordered cells tends to support the idea that asynchrony in the differentiation process is not the leading cause of our observed increase in entropy.

However the intrinsic burstiness of the gene expression process [24, 73–75] might cause some issues in the use of cell re-ordering algorithms. We therefore examined this question by using a more formal approach. We reasoned that a modeling strategy might be useful in establishing the role asynchrony might play, especially since forcing a synchronous differentiation is not accessible in vitro, but can be done in silico. We used a two-state model of gene expression [27, 39–41, 56], for which we could learn the parameters from the data (see Materials and Methods). In the synchronous case, we obtained a variation in entropy resembling the one we calculated from the data (Fig 9E). The introduction of asynchrony induced a flatter time profile of the entropy (Fig 9F).

This finding did not, however, prove that our cells are synchronously differentiating, but only demonstrated the effect of asynchrony: in the background of bursty gene transcriptional process, asynchrony will tend to smoothen (and not augment) the entropy of the system. Therefore the observed surge in entropy can not be attributed to the asynchrony of the process.

The rise-and-fall of entropy in our data is in line was examined in a different setting, namely a reprogramming process [58]. The authors stated, "The initial transcriptional response is relatively homogeneous," offering the opportunity to examine the entropy time profile in such a homogeneous process. Our analysis of this dataset produced a similar behavior for entropy which significantly increased initially, before returning to lower values (S8 Fig).

Altogether our analysis is compatible with the notion that the rise and fall in entropy is the consequence of the dynamical behavior of the underlying gene regulatory network.

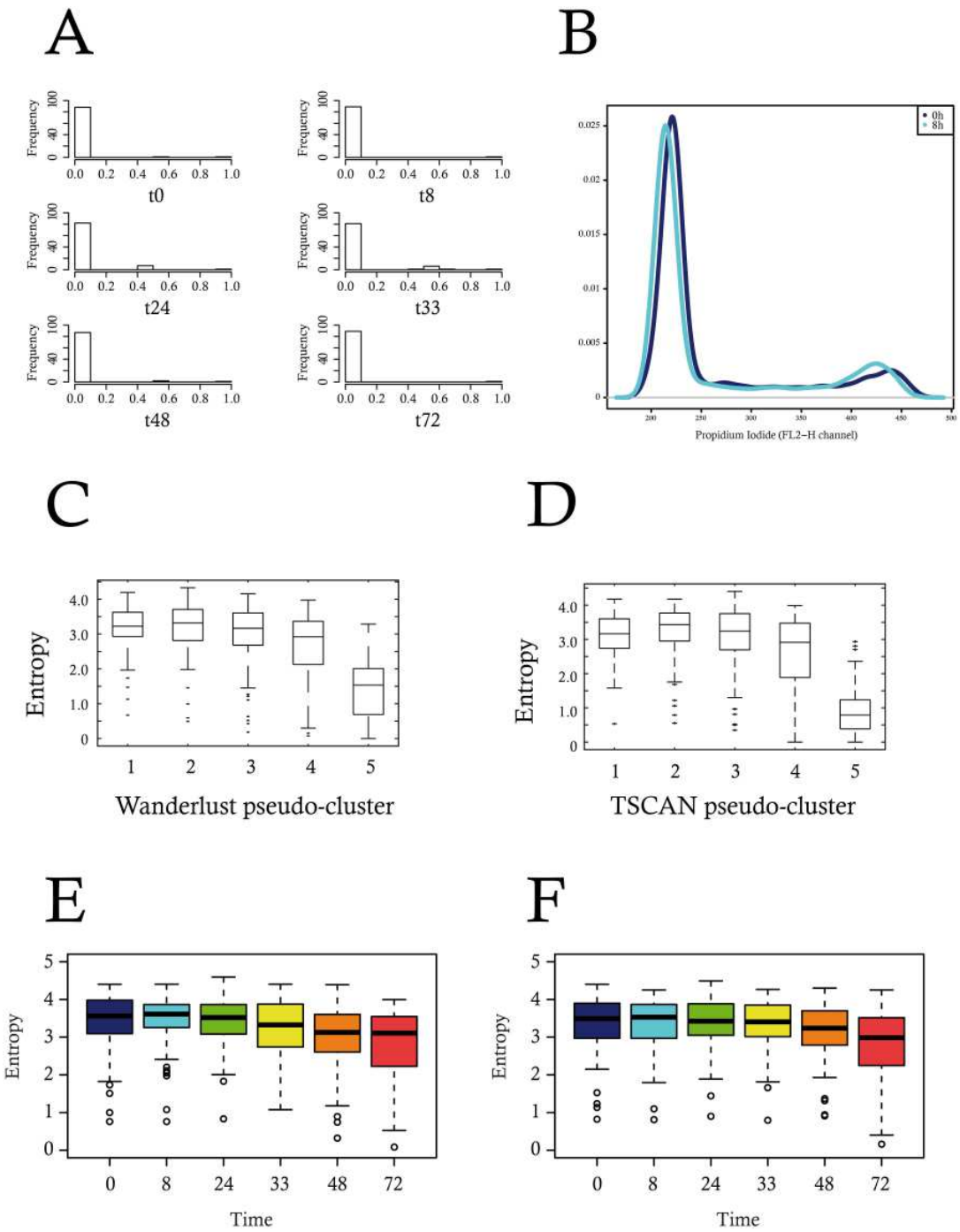


Fig 9. Exploration of potential cofounding factors. (A) Correlation of the CTCF gene with the rest of the 91 genes, at all six time-points. (B) FACS analysis of the cell cycle repartition at 0 and 8 h of differentiation. The difference between the two distributions was found not to be statistically significant ($p = 0.18$ using a Wilcoxon test). (C and D): calculation of the entropy content per cluster of cells re-organized using either WANDERLUST (C) or TSCAN algorithm (D). (E and F) In silico comparison of the effect of a synchronous versus an asynchronous differentiation process on the evolution of entropy. Data for this figure (C to F) can be found at osf.io/k2q5b.

doi:10.1371/journal.pbio.1002585.g009

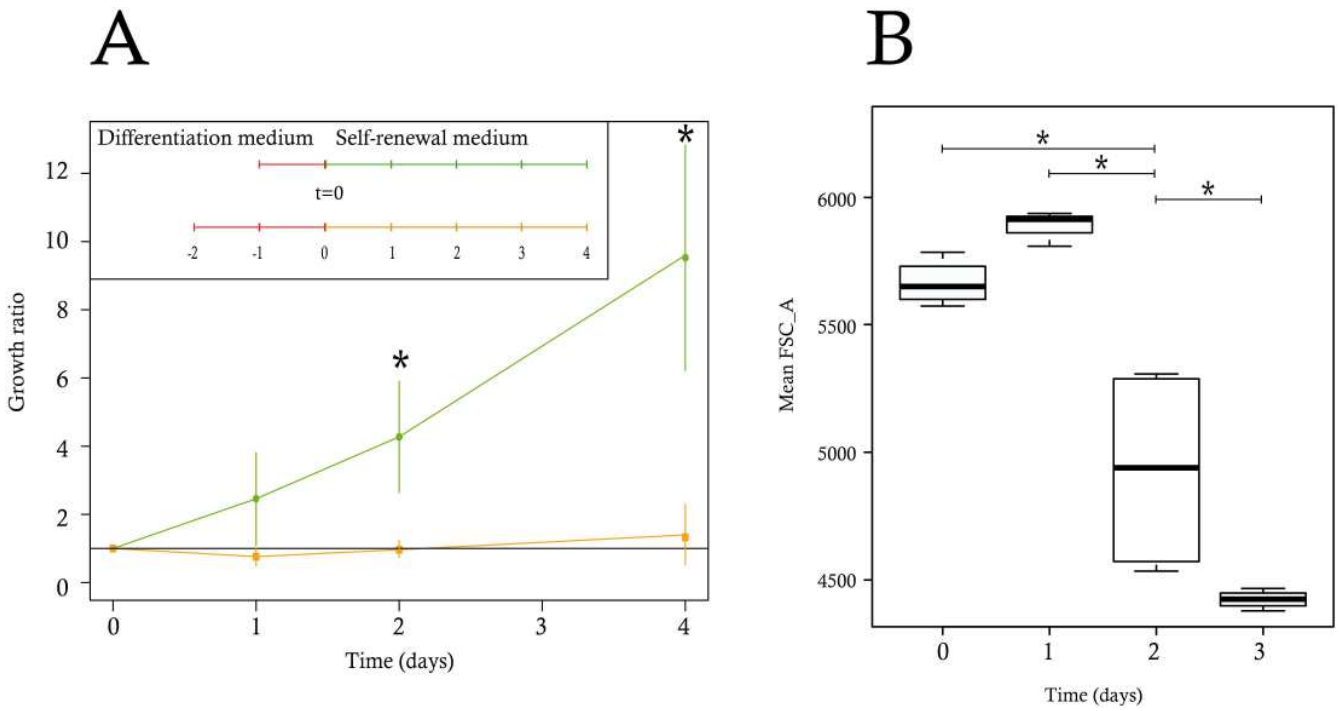


Fig 10. Evolution of physiological differentiation parameters. (A) T2EC were induced to differentiate for 24 and 48 h and subsequently seeded back in self-renewal conditions. Cells were then counted every day for 5 d. The green curve represents the growth of cells induced to differentiate for 24 h and the orange curve indicates the growth of cells induced to differentiate for 48 h. The data shown are the mean +/− standard deviation calculated on the basis of three independent experiments for the time-points 72 h and 96 h and four experiments for all other time-points. The growth ratio was computed as the cell number divided by the total cells at day 0. The significance of the difference between growth ratios at 24 h and 48 h was calculated using a Wilcoxon test. (B) The boxplots of the mean size observed were based on four independent experiments, each using 50,000 cells, using FSC_A as a proxy for cell size. All of the variances were compared by pairs using the F test and the * indicates when the variances were significantly different. Data for this figure can be found at osf.io/k2q5b.

doi:10.1371/journal.pbio.1002585.g010

The Point of No Return in T2EC Differentiation is Located between 24 h and 48 h

The above analysis of single-cell transcript profiles displays the following pattern:

1. A decrease in correlation value is observed between 0 and 8 h, and then correlation increases between 24 and 72 h.
2. An increase in I score value is observed between 0 and 8 h, then a return to its initial value at about 33 h, before continuing to decrease gradually.
3. A surge in entropy is significant at 8–24 h, and significantly decreases between 24 and 72 h.

Altogether, those results point toward the 8 and 24 h time-points as being a possible decision point, hence, a "point-of-no-return" in the differentiation process, beyond which cells are irreversibly committed toward erythrocytic differentiation. Consequently, we hypothesized that committed cells would be unable to revert back to a self-renewal process after 24 h of differentiation. To test this hypothesis we induced T2EC to differentiate for 24 h or 48 h, after which cells were transferred back into the self-renewal medium, in order to determine whether or not cells could revert back to the undifferentiated state after they had received differentiation signals for a given period of time. We observed that T2EC induced to differentiate for 24

h were still able to self-renew upon change of medium, while cells induced for 48 h could not do so (Fig 10A).

T2EC induced for 48 h seemed to stay in a quiescent state until they died. We therefore concluded that the physiological point of no return is located between 24 h and 48 h of our differentiation process, as suggested by our in silico analysis. Finally we determined whether cell size, a phenotypic integrated variable that has historically been used to monitor erythroid maturation [76, 77] would manifest the behavior of the underlying molecular network with respect to cell-cell variability. We therefore assessed cell size variation during the differentiation process. As expected [32], mean cell size started to decrease during differentiation to reach a minimum by 72 h (Fig 9B). Interestingly, cell size variability significantly peaked at 48 h before dropping precipitously by 72 h. Thus the high variability of gene expression observed at 24 h preceded a significant peak in cell size variability 1 d later.

Discussion

In the present work we assessed, using single-cell RT-qPCR, the temporal changes of gene expression in individual cells in a population of cells undergoing differentiation. For this, we used a physiologically relevant cellular system, which presents three main advantages: (i) those cells are primary, non-transformed cells; (ii) they do not show any tendency to spontaneous differentiation; and (iii) they can only differentiate along the erythrocytic lineage, excluding heterogeneity arising from coexistence of cells differentiating along different lineages.

To quantitatively assess the role of gene expression variability, we first defined a subset of genes relevant for analyzing the differentiation process. At the level of whole-population analysis this gene subset allowed a clear distinction among differentiation time-points. However, when assessed at the single-cell level, our analyses revealed a much higher cellular heterogeneity. Despite this heterogeneity, the selected genes were still effective in separating the two most extreme time-points in T2EC differentiation, confirming that information associated with the differentiation process is embodied in the gene expression data at the single-cell level. From the dataset that we generated at the single-cell level, two main results could be obtained: (i) regarding the biology of the erythroid differentiation, we identified previously unidentified genes as being important components of the self-renewal and differentiation of erythroid progenitors, and (ii) on a larger perspective, our results fully supported a dynamical view where differentiation can be seen as a critical phase transition driven by stochasticity.

Identification of new genes involved in the erythroid differentiation process

One question deals with the possible identification of important genes that can be seen as "drivers" of the process. At least three list of genes were generated during the course of this work that may qualify:

1. the "early drivers," genes identified in the wave analysis;
2. the genes qualifying for the DNB, and
3. the most densely connected genes in the correlation graph;

Restricting only to the most densely correlated genes at 0 and 8 h (since the two other lists were validated on those time-points), one observed a partial overlap between the three lists (S9 Fig), with no gene being common to all three lists. One possible explanation is simply that the three lists were obtained through different approaches, not supposed to identify the same set of genes. This result nevertheless suggests that although all of those genes might be functionally

important for the differentiation process, they might be involved in the global response at different levels. The early drivers might be more important for informing the whole network at early time points, whereas the two other genes sets might be involved in a more global reconfiguration of the network at later time-points. In any case those gene lists are to be seen as traces resulting from the behavior of the underlying dynamical network, and should not be mistaken for the dynamical network itself. It would therefore be of utmost importance to be able to correctly infer such a network. We are actively pursuing this goal in our group.

We discuss below possible functions of some of those genes, a full discussion for all genes being out of the scope of the present paper.

As previously mentioned, *Sca2* is a gene which we have previously shown to be associated with the self-renewal of erythroid progenitors [34].

LDHA encodes an enzyme that catalyzes the conversion of pyruvate to lactate, and has been involved in the Warburg effect (or anaerobic glycolysis), which is the propensity of cancer cells to take up glucose avidly and convert it to lactate [78]. Furthermore, deletion of *LDHA* has been shown to significantly inhibit the function of both hematopoietic stem and progenitor cells during murine hematopoiesis [79].

Since *LDHA* expression is under the control of HIF1α transcription factor [79], it could be involved in the response of immature erythroid progenitors to anemia. Those cells have to show a significant amount of self-renewal for recovering from a strong anemia, implying low oxygen condition [80]. It makes perfect sense that in this case the metabolism of self-renewing progenitors would rely upon an anaerobic pathway.

Moreover, HIF1alpha has also been shown to be an upstream regulator of HSP90*alpha* secretion in cancer cells in a protective way against the hypoxic tumoral environment [81]. Therefore, our results are in line with other findings showing that anaerobic glycolysis is favored in hypoxic conditions, such as the bone marrow environment, and required for stem cell maintenance [82]. Otherwise, since *LDHA* and *HSP90alpha* form part of the lists of potentially important genes between 0 and 8 h, our finding suggests that erythroid differentiation might be accompanied by a change from anaerobic glycolysis toward mitochondrial oxidative phosphorylation, as recently proposed [83].

Finally, our analysis highlighted the importance of the sterol synthesis pathway in the self renewal process since:

1. Among genes identified by RNAseq whose expression changed significantly, we found different genes associated to the sterol synthesis, such as *HMGCS1*, *CYP51A1*, *DHCR24*, *DHCR7*, *STARD4*, and *NSDHL* (S4 Fig);
2. The expression of those genes decreased promptly after the change of the external conditions, i.e the induction of the differentiation (Fig 7);
3. *STARD4* was both an early driver and one of the genes that displayed the highest number of edges at 0 h (Fig 5C). It has recently been demonstrated that *STARD4* expression could be used as poor prognosis gene in a six genes signature that defines aggressive subtypes in adult acute lymphoblastic leukemia [84].

These observations support the importance of sterol synthesis in the maintenance of cellular self renewal state and the necessity of a decrease of some sterol associated genes expression to allow the differentiation. The question as to why this group of genes act as the early sensors of change in environmental conditions remains elusive. In line with our previous results [35], one could hypothesize that cholesterol synthesis is a barrier toward differentiation/apoptosis that has to be lowered for differentiation to proceed.

A functional role for the surge in gene expression during critical transition?

On a more global perspective, the importance of cell-to-cell heterogeneity as a "biological observable" at the single-cell level, even among cells classified as belonging to the same "cell type" [85], is increasingly recognized [86]. But to what extent and when is such heterogeneity functionally important? Most single-cell transcript profile analyses of cell populations have so far focused mostly on computational descriptive analysis to identify clusters, and temporal progression, or to test dimensionality reduction and visualization tools, but less so to test a biological hypothesis. Here we used the single-cell granularity of gene expression analysis to test the long-standing hypothesis that stochastic cell-cell variability is not simply the byproduct of molecular noise but that such randomness of cell state plays a key role in differentiation [28]. In this Darwinian view, differentiation starts with an unstable gene expression pattern, generating cell type diversity. Therefore, one testable prediction was that an increase in gene expression heterogeneity should be observed during the critical phase of cell differentiation whenever the irreversible decision to commit is made.

Our main contribution is a demonstration that the increase in molecular variability precedes critical functional variations in cellular parameters, most importantly including the commitment status of the cells. Taken together, the timing of three observables achieved at single-cell resolution provides a coherent picture of a temporal structure of differentiation that would be invisible to traditional whole-population averaging techniques: (i) the surge in cell-to-cell variability of gene expression patterns of individual cells at 8–24 h; (ii) a sudden drop in the overall correlation, concomitant with the emergence of a DNB; and (iii) followed by the phenotypic marker of differentiation, the decrease of cell size, for which variability peaks at 48 h.

An important question is the relevance of that peak in variability. We demonstrated experimentally that no cell was able to return to a self-renewal state after 48 h in a differentiation medium. A similar timing for point-of-no return has previously been suggested in FDPC-mix cells [87]. Such an irreversible commitment to differentiation preceded by a highly significant increase in cell-to-cell variability is consistent with the explanation that cells differentiate by passing through two phases [87]: a first phase in which the self-renewing state is destabilized and primed by perturbation of their extracellular environment, followed by a second phase of a stochastic commitment to differentiation.

These observables (emergence of a DNB, drop in correlation, significant increase in entropy, surge in cellular parameters variations) jointly suggest a critical state transition, a class of dynamical behaviors that has been proposed to explain the qualitative, almost discrete and noise-driven "switching" into a new cell state as embodied by differentiation [88]. This conceptual framework naturally explains the irreversibility of fate commitment [89]. Indeed the maximum of the above three observables coincided with the functionally demonstrated point-of-no return to the self-renewal state in T2EC differentiation process, which was located between 24 and 48 h.

From a more biological perspective, we can view differentiation induction as a process of adaptation in which the cell's internal molecular network, adapted for growth in self-renewal conditions, has to adjust to the new external conditions when differentiation is induced by the change in external conditions. For example, in yeast, it has been shown that a nonspecific transcriptional response reflecting the natural plasticity of the regulatory network supports adaptation of cells to novel challenges [90]. The underlying mechanisms are yet to be discovered, but one would expect global mechanisms to be involved. Modifications of the chromatin dynamics [27] under the possible control of metabolic changes [91] are obvious candidates for such a role. Fluctuation in important transcription factor level has

also been proposed to be involved [92]. The surge of non-specific variability would allow exploration of new regions in the gene expression space. Preventing such an increase in variability has been associated to trapping cells in an undifferentiated state [93]. This increase would lead to a reconfiguration of the gene expression network into a state which is compatible with differentiation conditions and which is robust and consistent with a new attractor state in the network [29]. Then the decrease of molecular variability might reflect the implementation of the fully differentiated phenotype as cells settle down in the next stable state.

In this study, we exploited the wealth of information available in single-cell data by highlighting the critical molecular changes occurring along the differentiation sequence. First, the initial gene expression waves might represent a very early signal that happens between 0 and 8 h, followed by a pre-transition warning signal revealed by the DNB analysis, concomitant with the drop in gene correlations and the rise in cell-to-cell variability. Such a pattern are thought to reflect the underlying dynamical molecular mechanisms that drives the evolution of cells through the differentiation process. The first signals could be seen as an adaptative response to environmental changes, as suggested above, whereas the last warning signal, before irreversible commitment, could be seen as the point of cell decision making. At that stage it is hard to really be sure that the DNB genes actually drives the critical transition, but at the very least they represent a clear signal that our cells are experiencing such a transition. Until 24 h, at least, cells would still be able to functionally respond to self-renewal signals. This implies that at that stage the state of the network would be compatible with both a differentiation and a self-renewal process. One of the remaining challenging questions is what makes the cell takes the irreversible decision to differentiate at a point when the system seems to be totally disorganized. We strongly believe that this will be an emerging properties from the behavior of dynamical high-dimensional molecular network.

While the current study offers a single-cell resolution view on gene expression, it does so only through snapshots at strategically selected time-points. In the future it would therefore be of great importance to obtain a continuous measurement of the underlying gene expression network in order to explain the state changes in individual cells and to reconstruct the entire trajectory of each cell in gene expression state space. This information would expose the actual process of diversification that leads to the maximal heterogeneity marking the point of no return of differentiation.

NOTE ADDED IN PROOF: During the submission of this manuscript we became aware of the work of Mojtahedi, et al., 2016 (doi: 10.1371/journal.pbio.2000640) which arrived at a similar conclusion, and we cite that work in our discussion.

Materials and Methods

Cells and Culture Conditions

T2EC were extracted from bone marrow of 19-d-old SPAFAS white leghorn chickens embryos (INRA, Tours, France). These primary cells were maintained in self-renewal in LM1 medium (α-MEM, 10% Foetal bovine serum (FBS), 1 mM HEPES, 100 nM β-mercaptoethanol, 100 U/mL penicillin and streptomycin, 5 ng/mL TGF-α, 1 ng/mL TGF-β and 1 mM dexamethasone) as previously described [32]. T2EC were induced to differentiate by removing the LM1 medium and placing cells into the DM17 medium (α-MEM, 10% foetal bovine serum (FBS), 1 mM Hepes, 100 nM β-mercaptoethanol, 100 U/mL penicillin and streptomycin, 10 ng/mL insulin and 5% anemic chicken serum (ACS)). Differentiation kinetics were obtained by collecting cells at different times after the induction in differentiation.

Cell Population Growth Measurement

Cell population growth was evaluated by counting living cells using a Malassez cell and Trypan blue staining.

Propidium Iodide Staining

T2EC in self-renewal medium and T2EC induced to differentiate during 8 h were incubated for 30 min on ice with 100% cold ethanol, and then 30 min at 37˚C with 1 mg/mL RNase A (Invitrogen). Propidium Iodide (SIGMA) was added at 50 μg/mL 2 min prior to analysis and fluorescence was measured with the BD FacsCalibur 4-color flow cytometer, using the FL-2 channel. Data files were then extracted and analyzed using the bioconductor flowCore package.

T2EC Collection by Flow Cytometry

T2EC were collected individually in a 96-well plate using a flow cytometer (Facs ARIA I). Each individual cell was immediately gathered into a lysis buffer (Vilo [Invitrogen], 6U SUPERase-In [Ambion], 2.5% NP40 [ThermoScientific]), containing also Arraycontrol RNA spikes (Ambion). After collection, single-cells were immediately frozen on dry ice and stored at -80˚C.

Total RNA Extraction

Cell cultures were centrifuged and washed with 1X phosphate-buffered saline (PBS). Total RNA were extracted and purified using the RNeasy Plus Mini kit (Qiagen). Then, RNA were treated with DNAse (Ambion) and quantified using the Nanodrop 2000 spectrophotometer (Thermoscientific).

RNA-Seq Libraries Preparation

RNA-Seq libraries were prepared according to Illumina technology, using NEBNext mRNA library Prep Master Mix Set kit (New England Biolabs). Libraries were performed according to manufacturer's protocol. mRNA were purified using NEBNext Oligo d(T)25 magnetic beads and fragmented into 200 nucleotides RNA fragments by heating at 94˚C for 5 min, in the presence of RNA fragmentation Reaction Buffer. Fragmented mRNA were cleaned using RNeasy MinElute Spin Columns (Qiagen). Double strand cDNA were obtained by two-step RNA reverse transcription (RT) with random primers and purified using Magnetic Agencourt AMPure XP beads. To produce blunt ends, purified cDNA were incubated with NEBNext End Repair reaction buffer and NEBNext End Repair enzyme mix for 30 min at 20˚C. cDNA were purified again using Agencourt AMPure XP beads, and dA-tail were added to these cDNA fragments by incubating them with NEBNext dA-Tailing reaction buffer and klenow fragment for 30 min at 37˚C. After purification of the dA-tailed DNA, illumina adaptators were ligated to cDNA in the presence of NEBNext quick ligation reaction buffer, quick T4 DNA ligase, and USER enzyme. After size selection, purified adaptor-ligated cDNA were enriched by PCR with NEBNext High-fidelity 2X PCR Master mix, universal PCR primers and Index primers, and using thermal cycling conditions recommended by manufacturer's procedure. Finally, enriched cDNA were purified and sequenced by the Genoscope institute (Evry, France).

RNA-Seq Library Analysis

Sequencing files were loaded onto Galaxy (https://usegalaxy.org/). Quality was checked using FastQC. Groomed sequences were aligned on the galGal4 version of the chicken genome,

using TopHat [36]. The resulting .BAM files were transformed into .SAM files using SAM Tools. The gene counts table was generated using HTSeq [37] and the chr_M_Gallus_gallus .Galgal4.72.gtf annotated genome version. Differential gene expression was computed using `EdgeR` and plotted with the `plotSmear` function [38].

High-Throughput Microfluidic-based RT-qPCR

Every experiment related to high-throughput microfluidic-based RT-qPCR was performed according to Fluidigm's protocol (PN 68000088 K1, p.157–172) and recommendations.

Reverse transcription of isolated bulk-cell RNA and single-cell RNA.

- *Isolated bulk-cell RNA*
  Fifty nanograms of extracted bulk-cell RNA were reverse-transcribed using the Superscript III First-Strand Synthesis SuperMix for qRT-PCR kit (Invitrogen). The reverse transcription step and RNAse H treatments were performed according to manufacturer's instructions. Reverse transcription was performed during 30 min at 50˚C, followed by 5 min at 80˚C, and RNAse H treatment was run at 37˚C during 20 min. Finally, cDNA were stored at -20˚C.
- *Single-cell RNA*
  Single-cell lysates were thawed on ice and denatured for 1.5 min at 65˚C. RNA were reverse-transcribed in presence of SuperScript III Reverse Transcriptase enzyme, from the Super-Script VILO cDNA Synthesis kit (Invitrogen), and T4 gene 32 protein (New England Biolabs) to improve reverse transcription efficiency. The reaction thermal cycling conditions were 5 min at 25˚C, 30 min at 50˚C, 25 min at 55˚C, 5 min at 60˚C and 10 min at 70˚C.

Specific target amplification of cDNA. Primers were designed using the Ensembl database (http://www.ensembl.org/Gallus_Gallus/Info/Index/) and Primer3Plus software (http://www.bioinformatics.nl/primer3plus/). For information about the primers sequences used, please contact the authors.

The cDNA pre-amplification was performed using the TaqMan PreAmpMaster (Applied Biosystems) mixed with all primer pairs of the genes of interest (Sigma-Aldrich), diluted at 500 M. For single-cell cDNA pre-amplification, this reaction mix was also composed of 0.5 M pH8 EDTA. The thermal cycling program used for single-cell cDNA is 10 min of enzyme activation at 95˚C, followed by 22 cycles at 96˚C for 5 s and 60˚C for 4 min. For bulk-cell cDNA, the enzyme activation step was followed by 14 cycles at 95˚C for 15 s and 60˚C for 4 min.

Exonuclease treatment. Exonuclease I (*E. coli*, New England BioLabs) was used on pre-amplified cDNA to eliminate single-strand DNA. The treatment was performed at 37˚C during 30 min and then the enzyme was inactivated at 80˚C during 15 min. For bulk-cell, cDNA were diluted in TE (10 mM pH8 Tris, 1 mM EDTA). For single-cell, cDNA were diluted in low EDTA TE buffer (10 mM pH8 Tris, 100 nM EDTA). All samples were then stored at -20˚C.

RT-qPCR: data generation. Pre-amplified cDNA were mixed with Sso Fast EvaGreen Supermix With Low ROX (Bio-Rad) and DNA binding dye sample loading reagent (Fluidigm). Primer pairs of the genes of interest were diluted at 5 μM with the Assay Loading Reagent (Fluidigm) and low EDTA buffer. First, the 96.96 DynamicArray IFC chip (Fluidigm) was primed. Then, prepared cDNA and primer pairs were loaded in the inlets of this device.

To avoid chip-linked variability, when analyzing single-cell data we were careful to represent every time-point in each of the four microfluidic-based chip analyzed.

The prime step and transfer of cDNA samples and primers from the inlets into the chip were performed using the IFC Controller HX (BioMark HD system). The chip was analyzed

using the BioMark HD reader according to the GE 96 × 96 PCR + Melt v2.pcl program, thanks to the data collection software. Then, raw data were analyzed with the Fluidigm Real-Time PCR Analysis software.

Positive exogenous controls (RNA spikes) were used to validate the RT-qPCR experiment as recommended by Fluidigm Company. We also used the RNA spikes to normalize the data (see below). To determine qPCR efficiency of every primer pairs used, serial dilution scales of bulk-cell cDNA were performed. PCR efficiencies were calculated as follows: $E = 10^{-1/\text{slope}}$. Primer pairs presenting PCR efficiency less than 80% or more than 120% were removed from subsequent analyses.

RT-qPCR: low-level data analysis. First, a manual examination was performed regarding data quality. RTqPCR data were exported from the BioMark HD data collection software. On every microfluidic-based chip, each gene was controlled in a qualitative manner in order to keep only reliable and good quality data. For this we manually edited the data files by adding a new column named "DELETED." Numbers "0" or "1" were appended in this column according to various criteria. Quality control was based both upon amplification and melting curves examination. For one given gene all the melting curves had to be centered on a unique melting temperature. When a given melting curve peak shifted to a higher or lower Tm, "1" was added into the DELETED column for this amplification. Moreover, data displaying a double peak were also considered unreliable and annotated with a "1." Finally, "noisy" amplification curves departing from the smooth classical sigmoidal shape were also tagged as "1." We allowed the quantification cycle (Cq) to be as high as 30. For a higher number of cycles, the machine returned a value of 999, meaning that there were not enough molecules to be detected. After this quality control, Cq values of data tagged as "1" were replaced with UD (for "undefined") in the raw data file, since they would not be taken into account in later analysis. Then the new table underwent an automatic formatting consisting in a second multiple-criteria cleaning process using an in-house R script. Cq values were converted into (approximately) absolute numbers of molecules according to the following steps. First, we selected cells with at least one valid spike measurement (i.e., whose Cq is different from UD and 999). Then, we normalized the raw value $\widehat{\text{Cq}}_{i,j}$ for cell i and RNA j according to the cell mean spike value $\overline{\text{Cq}}_i$ (or the only available spike if one is invalid), with the global mean spike value $\overline{\text{Cq}}_0$ as reference. That is, the normalized value $\text{Cq}_{i,j}$ for cell i and RNA j is defined by

$$\text{Cq}_{i,j} = \widehat{\text{Cq}}_{i,j} - \left(\overline{\text{Cq}}_i - \overline{\text{Cq}}_0\right).$$

After removing cells with abnormally important amount of genes with low expression (high $\text{Cq}_{i,j}$ values, suggesting the absence of a cell in the well), the numbers of mRNA molecules were estimated, considering the following: a maximum Cq equal to 30 as the measurement of 1 molecule in the well after 22 cycles of pre-amplification, a dilution factor corresponding to 1 cell extract diluted in 96 wells, and a sampling of 1/45 for PCR measurement. Thus the number $\text{m}_{i,j}$ of RNA j molecules in cell i is given by

$$\text{m}_{i,j} = 96 \times 45 \times 2^{30-22-\text{Cq}_{i,j}}.$$

We consistently set $\text{m}_{i,j} = 0$ when $\text{Cq}_{i,j} = 999$, and $\text{m}_{i,j} = \text{UD}$ when $\text{Cq}_{i,j} = \text{UD}$.

Replacing missing values. Since some statistical tools (like PCA) do not support missing values, the UDs had to be replaced with some *appropriate* numerical values, i.e., that do not change the data distribution, nor introduce any artificial correlation.

To this end, we calibrated the marginal distribution of each gene at each time-point using the 3-parameter Poisson-Beta family, which corresponds to the stationnary distribution of the

widely-used "two-state" model of gene expression [39–41]. As emphasized in [41], it can be obtained as the mixture distribution $\mathcal{D}(a, b, c)$ of X resulting from the hierarchical model

$$\begin{cases} Z \sim \text{Beta}(a, b) \\ X \sim \mathcal{P}(cZ) \end{cases}$$

where a, b, and c are positive. Thus for each time-point t and each gene j, we estimated the parameters a_j^t, b_j^t and c_j^t by taking the absolute value of the moment-based estimators proposed in [39]. Note that these slightly modified estimators are also convergent since the parameters are assumed to be positive. This estimation was only performed for genes with at least 20 valid cells and conduced to delete genes with too many UDs. This led us to delete two genes, resulting in a total of 90 genes analysed. The data was fitted very well in practice, making it relevant to simply replace the UDs with independent samples from the corresponding distributions $\mathcal{D}(a_j^t, b_j^t, c_j^t)$. Considering the actual inferred parameter regime (large values of c, meaning that the numbers of molecules span a high range) and the continuous nature of our data, we actually ignored the Poisson step and sampled from $c_j^t \text{ Beta}(a_j^t, b_j^t) \approx \mathcal{D}(a_j^t, b_j^t, c_j^t)$.

Obviously, such artificially generated values should not be seen as data, but they ensure that the dimension-reduction algorithms perform at their best and compute relevant projection axes (e.g., the main two axes for a PCA). We checked that indeed consistent PCA outputs were generated from different UD replacement operations (not shown).

Technical Reproducibility

Since RT-qPCR experimental procedure introduces unavoidable technical noise, we decided to explore which steps were the main sources of this variability (S1 Fig). We first assessed the reproducibility of the cDNA pre-amplification step by amplifying four cDNA samples from the same RT before analyzing it by qPCR. Gene expression levels differences between pre-amplification replicates were found to be negligible (S1A and S1B Fig). We then checked the RT-qPCR amplification step by analyzing the *RPL22L1* gene three times per chip. Expression levels between *RPL22L1* triplicates were quantitatively extremely similar (S1C to S1E Fig), confirming that amplification brings a negligible amount of variability as previously shown [42, 43]. We also tested the experimental variability induced by the RT reaction. We observed significant gene expression level differences between three RT from the same sample (S1A and S1F Fig), contrary to replicates from other critical steps. Indeed, it has been demonstrated and discussed that the RT reaction is the main source of technical noise, since it introduces biases through priming efficiency, RNA integrity and secondary structures and reverse transcriptase dynamic range [42, 44, 45]. In order to estimate the amount of variation introduced in our experiments by this step, we used external RNA spikes. The variation affecting those spikes spanned 5.8 Cqs (mean of Cq_{max}–Cq_{min} across the spikes) whereas the variability affecting the genes spanned a much larger region of 22.9 Cqs (mean of Cq_{max}–Cq_{min} across the genes), showing that the biological variability was much larger than the variability introduced by the RT step.

Statistical Analysis

Software. Most of the statistical analyses were performed using R [46]. The *k*-means clustering was performed using the `stats` R library. PCAs were performed using the `ade4` package [47]. All PCAs were centered (mean subtraction) and normalized (dividing by the standard deviation). All PCAs were displayed according to PC1 and PC2, which are the first and second axis of the PCA respectively. Hierarchical cluster analysis was performed applying

the R `hclust` function, using the complete linkage method on Euclidean distances. Dendrograms were built and plotted using the `dendextend` R library. Correlation analysis was performed using `rcorr` from the `Hmisc` R library. The *p*-value was corrected for multiple testing using the Bonferroni method [48]. Networks were computed using Cytoscape [49]. Cross-correlation analysis was performed using the `matcor` function from the `CCA` R library. Normality of the distributions was tested using the `shapiro.test` function. The variances were compared using the F test with the `var.test` function. Wilcoxon test was performed using the `wilcox.test` function. t-SNE and diffusion maps were computed using the Matlab Toolbox for Dimensionality Reduction (http://lvdmaaten.github.io/drtoolbox/). The t-SNE analysis was performed on a normalized version of the data, using `zscore` function. Kernel PCA was computed using the Matlab `kPCA` script [50] applying polynomial with fractional power 0.1. All linear analysis methods (PCA, HCA and correlation analysis) were performed after applying the transformation $m \mapsto \ln(m + 1)$ to the data, which gives access to the more linear Cq structure. All non-linear analysis methods (t-SNE, diffusion maps and Kernel PCA) were performed using untransformed m values.

I score calculation. The I score was calculated as previously described in [51] as follows: among the $n = 90$ studied genes, we defined a subset D containing n_D genes. We then defined the I score as:

$$\mathrm{I} = \mathrm{CV}\frac{\mathrm{PCC}_{in}}{\mathrm{PCC}_{out}}$$

with

$$\mathrm{CV} = \frac{1}{n_D}\sum_{i\in D}\mathrm{CV}_i\,, \quad \mathrm{PCC}_{in} = \frac{1}{{n_D}^2}\sum_{i,j\in D}C_{i,j}\,, \quad \mathrm{PCC}_{out} = \frac{1}{n_D(n-n_D)}\sum_{\substack{i\in D\\ j\notin D}}C_{i,j}$$

where CV_i is the coefficient of variation of gene i and $C_{i,j}$ stands for Pearson's correlation coefficient between genes i and j.

Wave analysis. One thousand boot-strap expression matrices were generated from genes RNA counts distribution for each time-point (0, 2, 4, and 8 h). New expression matrices were generated by uniform sampling of cells, which correspond to matrix lines, using the `randsample` Matlab command with replacement. For each time-point combination, a Mann-Whitney U test was performed using the `ranksum` Matlab command to detect significant variation. Wave membership was based on time variations. By definition a gene belongs to the wave at time T if there is at least one variation detected between time T and a previous time-point and if the gene does not belong to a previous wave. Only genes identified in a wave that displayed a significant variation in more than 90% of boot-straped samples were kept in this wave.

Estimation of entropy. We estimated the Shannon entropy of each gene j at each time-point t as follows: we computed basic histograms of the genes with $N = N_c/2$ bins, where N_c is the number of cells, which provided the probabilities $p^t_{j,k}$ of each class k. Finally, the entropies were defined by

$$E^t_j = -\sum_{k=1}^{N} p^t_{j,k}\log_2(p^t_{j,k}).$$

When all cells express the same amount of a given gene, this gene's entropy will be null. On the contrary, the maximum value of entropy will result from the most variable gene expression level (S2 Fig).

Re-ordering algorithms. We performed the pseudotemporal ordering of cells using three different algorithms: SCUBA [52], WANDERLUST [53] and TSCAN [54]. SCUBA is a two-step cell-ordering algorithm, in which one first reduces the data dimensionality by using t-SNE [55] and then determines the principal curve in the low-dimensional projection. We applied SCUBA by reducing the data into 2-D using tSNE (perplexity = 30) and by adopting k-segments algorithm (maximal number of segments = 8) as the option for the principal curve analysis. Since the differentiation path estimated by SCUBA was undirected, we set *LDHA* as the anchor-gene/marker to define the beginning and the end of pseudotime.

In contrast, WANDERLUST is a non-branching trajectory detection algorithm [53]. The method estimates the pseudotimes by representing each single-cell as a node in an ensemble of k-nearest-neighbor graph, followed by assigning a trajectory for each graph. This trajectory is defined by connecting cells with similar gene expressions through the shortest path. To reinforce this path assembly, a set of cells is randomly chosen as waypoints. The final cell ordering corresponds to the average trajectories over the ensemble of graphs. Here, we adopted the cosine similarity distance function for the trajectory detection, in which the single cell with the maximum *LDHA* expression was used as the initial node. Each cell's pseudotime has a value normalized between 0 and 1, reflecting its position along the differentiation path. For the entropy calculation, we grouped the cells into five pseudo-clusters, by collecting cells within five evenly spaced pseudotime window between 0 and 1 (e.g., pseudo-cluster 1 contained cells with pseudotime between 0 and 0.2, pseudo-cluster 2 contained cells with pseudotime between 0.2 and 0.4, and so on).

Finally, TSCAN is a cluster-based minimum spanning tree ordering algorithm [54]. The algorithm begins with clustering cells according to the similarity in their gene expressions, and continues with building the minimum spanning tree (MST) connecting the centroids of these clusters. The pseudotime is calculated by projecting each single cell to the MST edges. The algorithm also implements a preprocessing step involving gene clustering and dimensional reduction in order to alleviate the effect of drop-out events [54]. The preprocessing of our data produced 36 gene clusters, on which we employed the independent component analysis (ICA) to obtain a 2-D projection. Finally, we applied TSCAN using five cell clusters to generate the cell pseudotimes.

We computed the entropy for each cluster of cells following the procedure described above.

In silico simulations of mRNA level for single cells. In silico results were generated using the two-state model of gene expression [27, 39–41, 56]. We first inferred a set of model parameters (Kon, Koff, S0, D0) specific to each gene and depending on time. For that we used an inference method based on moment analysis [39] from our single cell expression matrix allowing to estimate three of these parameters (Kon, Koff and S0). To estimate D0 (mRNA degradation rate) we used population data of mRNA decay kinetic using actinomycin D-treated T2EC (osf.io/k2q5b). To simulate mRNA level we used the Gillespie algorithm [57]. In order to validate this modeling approach, we simulated for a given gene its mRNA evolution for 100 cells and extracted its distribution among cells at different time-points (0, 8, 24, 33, 48, and 72 h). We then compared in vitro and in silico distributions with a non-parametric Mann-Whitney U test. In silico measurements reproduced qualitatively the evolution of mean and distributions measured in vitro (not shown).

In silico simulations of the differentiation process. In order to stabilize the model before differentiation start, we ran the simulation for 100 h (model time) with constant parameters (value corresponding to 0 h). In silico differentiation was induced by a change in parameters values to now impose the parameters deduced from the in vitro data at different time-points. At each time step we computed parameters value with a linear interpolation between the two nearest time-points. For example at simulation time 4 h parameters

values correspond to the mean value between 0 and 8 h. We simulated 100 cells at each time-point. In order to study the impact of asynchronous differentiation, we compared two situations:

1. All cells had their parameters changed simultaneously, corresponding to a synchronous differentiation.
2. We randomly chose for each cell a time lag from a uniform distribution between 0 and 24 h. Then during the simulation, parameters started to change at t = 0 h + time lag. This corresponded to an asynchronous differentiation.

We then used the same metrics for analyzing those in silico distributions as those used for analyzing the in vitro data.

scRNA-seq data analysis. Counting table from [58] was downloaded from the following URL: http://www.ncbi.nlm.nih.gov/geo/download/?acc=GSE67310. The original (Log2 [FKPM]) data were transformed into FKPM data for analysis using the BPglm algorithm [59]. Running the algorithm with an FDR value of less than 0.00005 and using the Bonferroni correction method for multiple testing led us to a list of 776 differentially expressed genes, on which entropy was computed. Statistical significance was computed using the Wilcoxon non parametric test.

Supporting Information

S1 Fig. Reproducibility of the pre-amplification and RT-qPCR amplification steps. (A) the protocol used for assessing variation sources; (B) variations induced by four independent pre-amplifications when assessing the level of expression of the OSC gene; (C–E) variations induced by the PCR amplification step. The *RPL22L1* gene expression was analyzed three times per single-cell. Shown is the correlation between those three RT-qPCR replicates. The corresponding correlation coefficients are plotted on the graphs. The slopes of the linear regression lines are 0.99 for all three comparisons; (F) variations induced by three independent reverse-transcriptions when assessing the level of expression of the *OSC* gene.
(PDF)

S2 Fig. Schematic description of the entropy value. On the left are shown gene expression values that are transformed into probabilities (pj) to observe a given expression level in a cell population. The upper case illustrates the deterministic case where all cells do express the same expression level, resulting in a probability of 1 of observing such a level. This results in a null entropy (see Materials and Methods for the calculation). The lower case illustrates the other extreme case, where all the cells have different expression level, resulting in a much higher entropy.
(PDF)

S3 Fig. Scatter and MA plots showing the reproducibility of read counts between replicates and the differential expression during the differentiation process. (A,B) Relationship between biological replicates of two independent RNA-Seq experiments: self-renewing T2EC (left panel) and T2EC induced to differentiate for 48 h (right panel). For each condition, the *x*-axis represents the read counts of the first biological experiment, whereas read counts of the second biological replicate are given on the *y*-axis. Each dot corresponds to the expression level of one gene. (C) Comparative analysis of RNA-Seq data generated from two independent libraries of T2EC in self-renewing state and T2EC induced to differentiate for 48 h. The *x*-axis shows the expression level of each gene (transcript raw counts divided by the library size and multiplied by 1 million, averaged between the two independent libraries)

while the fold change (self-renewal versus differentiation) appears in the *y*-axis. Red-colored dots highlights genes that are significantly differentially expressed (p-value < 0.05).
(PDF)

S4 Fig. Identification of common patterns of expression during the differentiation process using K-means clustering. K-means clustering was used to separate the 110 selected genes into seven clusters regarding the expression profiles along the differentiation process. Starting models of gene expression pattern, corresponding to the centroid of each cluster, are represented in the first graph (starting cluster). We identified seven patterns of gene expressions with increasing, decreasing and one complex (cluster 4) dynamic profiles. The final centroid was recalculated after gene allotment, and might slightly differ from the starting one.
(PDF)

S5 Fig. Representation of the 92 selected genes. (A) On the basis of RNA-Seq data and k-means analysis (S4 Fig), the 92 genes selected for the single-cell analysis (S1 Table) can be separated into three types: up-regulated (red circles), invariant (green circles), and down-regulated genes (blue circles) at 48 h of the differentiation process. For each gene (*x*-axis) the fold-change (FC) between the self-renewal state and the differentiation state at 48 h (Diff/SR) was plotted along the *y*-axis. (B) Representation of known connections among the 92 genes selected according to the STRING database (http://string.embl.de/). Each edge between two genes corresponds to a known association between those genes. The densely connected component at the center of the network graph is composed of genes involved in sterol biosynthesis. A cluster of gene encoding porteins involved in signal transduction is apparent on the top right part of the network.
(PDF)

S6 Fig. Cross-correlation analysis between the gene expression value in populations and in single cells. The correlation matrix is divided into four smaller matrices: the correlation matrix of each dataset (populations: top-left panel; single-cells: bottom-right panel) and the correlation matrix between the two datasets (top-right and bottom-left panels, showing the same values). The values of the correlations are color-coded according to the scale given below. Correlation are calculated for each gene either accross populations samples or across single cells.
(PDF)

S7 Fig. Distributions of the expression values for three genes up-, down-, and non-regulated during the differentiation process. The histograms show the expression distribution of three genes among single cells at 0 and 72 h differentiation time-points. The gene expression levels (m value) are shown on the *x*-axis, the number of cells (count) is represented on the *y*-axis.
(PDF)

S8 Fig. Variation of entropy during a reprogramming process. We computed differential gene expression between 0 and 2 d using the scRNA-seq data from [58]. We then computed an entropy value per time-point for the 776 resulting genes. Statistical significance was computed using a Wilcoxon test.
(PDF)

S9 Fig. Overlapping genes between DNBs, early drivers and correlation network nodes at 0–8 h of differentiation. The Venn diagram shows the overlap of the three lists of genes obtained from the initial expression waves analysis (green), the correlation networks (pink), and the DNB theory (blue). The common genes between these lists were searched at 0 and 8 h

when all three analyses have been performed (early driver genes were only identified between 0 and 8 h).
(PDF)

S1 Table. Supplementary Table 1. Shown is the complete list of the 92 genes we analyzed, together with their expression value in the four RNA-Seq libraries (SR_1 and SR_2 being the two independent libraries made using self-renewing cells and Diff_1 and Diff_2 being two independent libraries made from cells differentiated for 48 h) and the group of variation at 48 h to which they belong (up-, down-, or non-regulated).
(CSV)

Acknowledgments

We would like to thank the following colleagues for helpful advice and discussions throughout this work: Vincent Lacroix (LBBE/UCBL), Marie Sémon (IGFL/ENSL), Gaël Yvert (LBMC/ENSL), Didier Auboeuf (LBMC/ENSL), Isabelle Durand (CLB), David Cox (CLB), Nicole Dalla-Venezia (CLB), Jordan C. Moore (Fluidigm), Frédéric Moret (INMG/UCBL), Julien Falk (INMG/UCBL), and all the members of the Stochagène and Iceberg projects. We thank the Génoscope, and especially Carole Dossat, for their invaluable help in sequencing of the RNAseq libraries. We would like to thank Gérard Benoit (LBMC/ENSL), Marieke von Lindern (Sanquin, Amsterdam), and Sui Huang (ICSB, Seattle) for critical reading of the manuscript. We thank Geneviève Fourel for pointing our attention to the [58] paper.

Author Contributions

Conceptualization: OG UH AB AR SGG JJK NPG RG JC.

Formal analysis: AR LB UH AB NPG RG TE OG.

Funding acquisition: OG SGG.

Investigation: AR VM EV AG OA.

Supervision: OG SGG.

Writing – original draft: AR SGG OG.

Writing – review & editing: OG AR SGG JJK LB JC RG NPG.

References

1. Wolff L, Humeniuk R. Concise review: erythroid versus myeloid lineage commitment: regulating the master regulators. Stem Cells. 2013; 31(7):1237–44. doi: 10.1002/stem.1379 PMID: 23559316
2. Torres-Padilla ME, Chambers I. Transcription factor heterogeneity in pluripotent stem cells: a stochastic advantage. Development. 2014; 141(11):2173–81. doi: 10.1242/dev.102624 PMID: 24866112
3. Singer ZS, Yong J, Tischler J, Hackett JA, Altinok A, Surani MA, et al. Dynamic heterogeneity and DNA methylation in embryonic stem cells. Mol Cell. 2014; 55(2):319–31. doi: 10.1016/j.molcel.2014.06.029 PMID: 25038413
4. Luo Y, Lim CL, Nichols J, Martinez-Arias A, Wernisch L. Cell signalling regulates dynamics of Nanog distribution in embryonic stem cell populations. J R Soc Interface. 2012; doi: 10.1098/rsif.2012.0525
5. Chickarmane V, Olariu V, Peterson C. Probing the role of stochasticity in a model of the embryonic stem cell: heterogeneous gene expression and reprogramming efficiency. BMC Syst Biol. 2012; 6:98. doi: 10.1186/1752-0509-6-98 PMID: 22889237

6. Sturrock M, Hellander A, Matzavinos A, Chaplain MA. Spatial stochastic modelling of the Hes1 gene regulatory network: intrinsic noise can explain heterogeneity in embryonic stem cell differentiation. J R Soc Interface. 2013; 10(80):20120988. doi: 10.1098/rsif.2012.0988 PMID: 23325756
7. Ochiai H, Sugawara T, Sakuma T, Yamamoto T. Stochastic promoter activation affects Nanog expression variability in mouse embryonic stem cells. Sci Rep. 2014; 4:7125. doi: 10.1038/srep07125 PMID: 25410303
8. Wu J, Tzanakakis ES. Deconstructing stem cell population heterogeneity: Single-cell analysis and modeling approaches. Biotechnol Adv. 2013; 31:1047–1062. doi: 10.1016/j.biotechadv.2013.09.001 PMID: 24035899
9. Buganim Y, Faddah DA, Cheng AW, Itskovich E, Markoulaki S, Ganz K, et al. Single-cell expression analyses during cellular reprogramming reveal an early stochastic and a late hierarchic phase. Cell. 2012; 150(6):1209–22. doi: 10.1016/j.cell.2012.08.023 PMID: 22980981
10. Haas S, Hansson J, Klimmeck D, Loeffler D, Velten L, Uckelmann H, et al. Inflammation-Induced Emergency Megakaryopoiesis Driven by Hematopoietic Stem Cell-like Megakaryocyte Progenitors. Cell Stem Cell. 2015; doi: 10.1016/j.stem.2015.07.007 PMID: 26299573
11. Pina C, Fugazza C, Tipping AJ, Brown J, Soneji S, Teles J, et al. Inferring rules of lineage commitment in haematopoiesis. Nat Cell Biol. 2012; 14(3):287–94. doi: 10.1038/ncb2442 PMID: 22344032
12. Kouno T, de Hoon M, Mar JC, Tomaru Y, Kawano M, Carninci P, et al. Temporal dynamics and transcriptional control using single-cell gene expression analysis. Genome Biol. 2013; 14(10):R118. doi: 10.1186/gb-2013-14-10-r118 PMID: 24156252
13. Feinerman O, Veiga J, Dorfman JR, Germain RN, Altan-Bonnet G. Variability and robustness in T cell activation from regulated heterogeneity in protein levels. Science. 2008; 321(5892):1081–4. doi: 10.1126/science.1158013 PMID: 18719282
14. Shalek AK, Satija R, Adiconis X, Gertner RS, Gaublomme JT, Raychowdhury R, et al. Single-cell transcriptomics reveals bimodality in expression and splicing in immune cells. Nature. 2013; 498 (7453):236–40. doi: 10.1038/nature12172 PMID: 23685454
15. Arsenio J, Kakaradov B, Metz PJ, Kim SH, Yeo GW, Chang JT. Early specification of CD8+ T lymphocyte fates during adaptive immunity revealed by single-cell gene-expression analyses. Nat Immunol. 2014; 15(4):365–72. doi: 10.1038/ni.2842 PMID: 24584088
16. Buettner F, Natarajan KN, Casale FP, Proserpio V, Scialdone A, Theis FJ, et al. Computational analysis of cell-to-cell heterogeneity in single-cell RNA-sequencing data reveals hidden subpopulations of cells. Nat Biotechnol. 2015; 33(2):155–60. doi: 10.1038/nbt.3102 PMID: 25599176
17. Helmstetter C, Flossdorf M, Peine M, Kupz A, Zhu J, Hegazy AN, et al. Individual T helper cells have a quantitative cytokine memory. Immunity. 2015; 42(1):108–22. doi: 10.1016/j.immuni.2014.12.018 PMID: 25607461
18. Lu Y, Xue Q, Eisele MR, Sulistijo ES, Brower K, Han L, et al. Highly multiplexed profiling of single-cell effector functions reveals deep functional heterogeneity in response to pathogenic ligands. Proc Natl Acad Sci U S A. 2015; 112(7):E607–15. doi: 10.1073/pnas.1416756112 PMID: 25646488
19. Elowitz MB, Levine AJ, Siggia ED, Swain PS. Stochastic gene expression in a single cell. Science. 2002; 297(5584):1183–1186. doi: 10.1126/science.1070919 PMID: 12183631
20. Suter DM, Molina N, Gatfield D, Schneider K, Schibler U, Naef F. Mammalian genes are transcribed with widely different bursting kinetics. Science. 2011; 332(6028):472–4. doi: 10.1126/science.1198817 PMID: 21415320
21. Kaern M, Elston TC, Blake WJ, Collins JJ. Stochasticity in gene expression: from theories to phenotypes. Nat Rev Genet. 2005; 6(6):451–64. doi: 10.1038/nrg1615 PMID: 15883588
22. Raj A, van den Bogaard P, Rifkin SA, van Oudenaarden A, Tyagi S. Imaging individual mRNA molecules using multiple singly labeled probes. Nat Methods. 2008; 5(10):877–9. doi: 10.1038/nmeth.1253 PMID: 18806792
23. Lestas I, Vinnicombe G, Paulsson J. Fundamental limits on the suppression of molecular fluctuations. Nature. 2010; 467(7312):174–8. doi: 10.1038/nature09333 PMID: 20829788
24. Viñuelas J, Kaneko G, Coulon A, Beslon G, Gandrillon O. Toward experimental manipulation of stochasticity in gene expression. Progress in Biophysics and Molecular Biology. 2012; 110:44–53. doi: 10.1016/j.pbiomolbio.2012.04.010 PMID: 22609563
25. Huh D, Paulsson J. Non-genetic heterogeneity from stochastic partitioning at cell division. Nat Genet. 2011; 43(2):95–100. doi: 10.1038/ng.729 PMID: 21186354
26. Cagatay T, Turcotte M, Elowitz MB, Garcia-Ojalvo J, Suel GM. Architecture-dependent noise discriminates functionally analogous differentiation circuits. Cell. 2009; 139(3):512–22. doi: 10.1016/j.cell.2009.07.046 PMID: 19853288

27. Viñuelas J, Kaneko G, Coulon A, Vallin E, Morin V, Mejia-Pous C, et al. Quantifying the contribution of chromatin dynamics to stochastic gene expression reveals long, locus-dependent periods between transcriptional bursts. BMC Biology. 2013; 11:15. doi: 10.1186/1741-7007-11-15 PMID: 23442824

28. Kupiec JJ. A Darwinian theory for the origin of cellular differentiation. Mol Gen Genet. 1997; 255 (2):201–8. doi: 10.1007/s004380050490 PMID: 9236778

29. Huang S. Systems biology of stem cells: three useful perspectives to help overcome the paradigm of linear pathways. Philosophical transactions Series A, Mathematical, physical, and engineering sciences. 2011; 366:2247–59.

30. Yvert G. 'Particle genetics': treating every cell as unique. Trends Genet. 2014; 30(2):49–56. doi: 10.1016/j.tig.2013.11.002 PMID: 24315431

31. Rebhahn JA, Deng N, Sharma G, Livingstone AM, Huang S, Mosmann TR. An animated landscape representation of CD4(+) T-cell differentiation, variability, and plasticity: Insights into the behavior of populations versus cells. Eur J Immunol. 2014; 44(8):2216–29. doi: 10.1002/eji.201444645 PMID: 24945794

32. Gandrillon O, Schmidt U, Beug H, Samarut J. TGF-beta cooperates with TGF-alpha to induce the self-renewal of normal erythrocytic progenitors: evidence for an autocrine mechanism. Embo J. 1999; 18 (10):2764–2781. doi: 10.1093/emboj/18.10.2764 PMID: 10329623

33. Damiola F, Keime C, Gonin-Giraud S, Dazy S, Gandrillon O. Global transcription analysis of immature avian erythrocytic progenitors: from self-renewal to differentiation. Oncogene. 2004; 23:7628–7643. doi: 10.1038/sj.onc.1208061 PMID: 15378009

34. Bresson C, Gandrillon O, Gonin-Giraud S. sca2: a new gene involved in the self-renewal of erythroid progenitors. Cell Proliferation. 2008; 41:726–738. doi: 10.1111/j.1365-2184.2008.00554.x

35. Mejia-Pous C, Damiola F, Gandrillon O. Cholesterol synthesis-related enzyme oxidosqualene cyclase is required to maintain self-renewal in primary erythroid progenitors. Cell Prolif. 2011; 44(5):441–52. doi: 10.1111/j.1365-2184.2011.00771.x PMID: 21951287

36. Trapnell C, Roberts A, Goff L, Pertea G, Kim D, Kelley DR, et al. Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks. Nat Protoc. 2012; 7(3):562–78. doi: 10.1038/nprot.2012.016 PMID: 22383036

37. Anders S, Pyl PT, Huber W. HTSeq-a Python framework to work with high-throughput sequencing data. Bioinformatics. 2014; doi: 10.1093/bioinformatics/btu638 PMID: 25260700

38. Robinson MD, McCarthy DJ, Smyth GK. edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. Bioinformatics. 2010; 26(1):139–40. doi: 10.1093/bioinformatics/btp616 PMID: 19910308

39. Peccoud J, Ycart B. Markovian Modelling of Gene Product Synthesis. Theoretical population biology. 1995; 48:222–234. doi: 10.1006/tpbi.1995.1027

40. Shahrezaei V, Swain PS. Analytical distributions for stochastic gene expression. Proc Natl Acad Sci U S A. 2008; 105(45):17256–61. doi: 10.1073/pnas.0803850105 PMID: 18988743

41. Kim JK, Marioni JC. Inferring the kinetics of stochastic gene expression from single-cell RNA-sequencing data. Genome Biol. 2013; 14(1):R7. doi: 10.1186/gb-2013-14-1-r7 PMID: 23360624

42. Stahlberg A. Comparaison os reverse transcriptases in gene expression analysis. clin Chem. 2004; 50:1678–80. doi: 10.1373/clinchem.2004.035469 PMID: 15331507

43. Pfaffl MW. A new mathematical model for relative quantification in real-time RT-PCR. Nucleic Acids Res. 2001; 29(9):e45. doi: 10.1093/nar/29.9.e45 PMID: 11328886

44. Brooks EM, Sheflin LG, Spaulding SW. Secondary structure in the 3' UTR of EGF and the choice of reverse transcriptases affect the detection of message diversity by RT-PCR. BioTechniques. 1995; 19:806–815. PMID: 8588921

45. Bustin SA. Quantification of mRNA using real-time reverse transcription PCR (RT- PCR): trends and problems. J Mol Endocrinol. 2002; 29:23–39. doi: 10.1677/jme.0.0290023 PMID: 12200227

46. Team RDC. R: A language and environment for statistical computing. Vienna, Austria ISBN 3-900051-07-0, URL http://wwwR-projectorg. 2008;.

47. Dray S, Dufour AB. The ade4 package: implementing the duality diagram for ecologists. Journal of Statistical Software. 2007; 22:1–20.

48. Bland JM, Altman DG. Multiple significance tests: the Bonferroni method. BMJ. 1995; 310(6973):170. doi: 10.1136/bmj.310.6973.170 PMID: 7833759

49. Shannon P, Markiel A, Ozier O, Baliga NS, Wang JT, Ramage D, et al. Cytoscape: a software environment for integrated models of biomolecular interaction networks. Genome Res. 2003; 13(11):2498–504. doi: 10.1101/gr.1239303 PMID: 14597658

50. Wang Q. Kernel Principal Component Analysis and its Applications in Face Recognition and Active Shape Models.; 2012.

51. Liu R, Chen P, Aihara K, Chen L. Identifying early-warning signals of critical transitions with strong noise by dynamical network markers. Sci Rep. 2015; 5:17501. doi: 10.1038/srep17501 PMID: 26647650

52. Marco E, Karp RL, Guo G, Robson P, Hart AH, Trippa L, et al. Bifurcation analysis of single-cell gene expression data reveals epigenetic landscape. Proc Natl Acad Sci U S A. 2014; 111(52):E5643–50. doi: 10.1073/pnas.1408993111 PMID: 25512504

53. Bendall SC, Davis KL, Amir el AD, Tadmor MD, Simonds EF, Chen TJ, et al. Single-cell trajectory detection uncovers progression and regulatory coordination in human B cell development. Cell. 2014; 157 (3):714–25. doi: 10.1016/j.cell.2014.04.005 PMID: 24766814

54. Ji Z, Ji H. TSCAN: Pseudo-time reconstruction and evaluation in single-cell RNA-seq analysis. Nucleic Acids Res. 2016; doi: 10.1093/nar/gkw430

55. Van der Maaten L, Hinton G. Visualizing data using t-SNE. Journal of Machine Learning Research. 2008; 9:2579–2605.

56. Paulsson J. Models of stochastic gene expression. Phys Life Rev. 2005; 2:157–175. doi: 10.1016/j.plrev.2005.03.003

57. Gillespie DT. A General Method for Numerically Simulating the Stochastic Time Evolution of Coupled Chemical Reactions. Journal of Computational Physics. 1976; 22(4):403–434. doi: 10.1016/0021-9991(76)90041-3

58. Treutlein B, Lee QY, Camp JG, Mall M, Koh W, Shariati SA, et al. Dissecting direct reprogramming from fibroblast to neuron using single-cell RNA-seq. Nature. 2016; 534(7607):391–5. doi: 10.1038/nature18323 PMID: 27281220

59. Vu TN, Wills QF, Kalari KR, Niu N, Wang L, Rantalainen M, et al. Beta-Poisson model for single-cell RNA-seq data analyses. Bioinformatics. 2016; doi: 10.1093/bioinformatics/btw202 PMID: 27153638

60. Dennis J G, Sherman BT, Hosack DA, Yang J, Gao W, Lane HC, et al. DAVID: Database for Annotation, Visualization, and Integrated Discovery. Genome Biol. 2003; 4(5):P3. doi: 10.1186/gb-2003-4-5-p3

61. Tian WL, He F, Fu X, Lin JT, Tang P, Huang YM, et al. High expression of heat shock protein 90 alpha and its significance in human acute leukemia cells. Gene. 2014; 542(2):122–8. doi: 10.1016/j.gene.2014.03.046 PMID: 24680776

62. Dong H, Zou M, Bhatia A, Jayaprakash P, Hofman F, Ying Q, et al. Breast Cancer MDA-MB-231 Cells Use Secreted Heat Shock Protein-90alpha (Hsp90alpha) to Survive a Hostile Hypoxic Environment. Sci Rep. 2016; 6:20605. doi: 10.1038/srep20605 PMID: 26846992

63. Haghverdi L, Buettner F, Theis FJ. Diffusion maps for high-dimensional single-cell analysis of differentiation data. Bioinformatics. 2015; doi: 10.1093/bioinformatics/btv325

64. Liu C. Gabor-Based Kernel PCA with Fractional Power Polynomial Models for Face Recognition. IEEE transactions on pattern analysis and machine intelligence. 2004; 26:572–581. doi: 10.1109/TPAMI.2004.1273927 PMID: 15460279

65. Piras V, Selvarajoo K. Reduction of gene expression variability from single cells to populations follows simple statistical laws. Genomics. 2014; PMID: 25554103

66. Pina C, Teles J, Fugazza C, May G, Wang D, Guo Y, et al. Single-Cell Network Analysis Identifies DDIT3 as a Nodal Lineage Regulator in Hematopoiesis. Cell Rep. 2015; 11(10):1503–10. doi: 10.1016/j.celrep.2015.05.016 PMID: 26051941

67. Singh A. Cell-Cycle Control of Developmentally Regulated Transcription Factors Accounts for Heterogeneity in Human Pluripotent Cells. Stem cell reports. 2013; 1:532–44. doi: 10.1016/j.stemcr.2013.10.009 PMID: 24371808

68. Swain PS, Elowitz MB, Siggia ED. Intrinsic and extrinsic contributions to stochasticity in gene expression. Proc Natl Acad Sci U S A. 2002; 99(20):12795–800. doi: 10.1073/pnas.162041399 PMID: 12237400

69. Padovan-Merhar O, Nair GP, Biaesch AG, Mayer A, Scarfone S, Foley SW, et al. Single mammalian cells compensate for differences in cellular volume and DNA copy number through independent global transcriptional mechanisms. Mol Cell. 2015; 58(2):339–52. doi: 10.1016/j.molcel.2015.03.005 PMID: 25866248

70. Chang HH, Hemberg M, Barahona M, Ingber DE, Huang S. Transcriptome-wide noise controls lineage choice in mammalian progenitor cells. Nature. 2008; 453(7194):544–7. doi: 10.1038/nature06965 PMID: 18497826

71. Klenova EM, Fagerlie S, Filippova GN, Kretzner L, Goodwin GH, Loring G, et al. Characterization of the chicken CTCF genomic locus, and initial study of the cell cycle-regulated promoter of the gene. J Biol Chem. 1998; 273(41):26571–9. doi: 10.1074/jbc.273.41.26571 PMID: 9756895

72. Suel GM, Kulkarni RP, Dworkin J, Garcia-Ojalvo J, Elowitz MB. Tunability and noise dependence in differentiation dynamics. Science. 2007; 315(5819):1716–9. doi: 10.1126/science.1137455 PMID: 17379809

73. Eldar A, Elowitz MB. Functional roles for noise in genetic circuits. Nature. 2010; 467(7312):167–73. doi: 10.1038/nature09326 PMID: 20829787

74. Larson DR, Singer RH, Zenklusen D. A single molecule view of gene expression. Trends Cell Biol. 2009; 19(11):630–7. doi: 10.1016/j.tcb.2009.08.008 PMID: 19819144

75. Senecal A, Munsky B, Proux F, Ly N, Braye FE, Zimmer C, et al. Transcription Factors Modulate c-Fos Transcriptional Bursts. Cell Rep. 2014; 8(1):75–83. doi: 10.1016/j.celrep.2014.05.053 PMID: 24981864

76. Dolznig H, Bartunek P, Nasmyth K, Müllner EW, Beug H. Terminal differentiation of normal erythroid progenitors: shortening of G1 correlates with loss of D-cyclin/cdk4 expression and altered cell size control. Cell Growth Differ. 1995; 6:1341–1352. PMID: 8562472

77. von Lindern M, Deiner EM, Dolznig H, Parren-Van Amelsvoort M, Hayman MJ, Mullner EW, et al. Leukemic transformation of normal murine erythroid progenitors: v- and c-ErbB act through signaling pathways activated by the EpoR and c-Kit in stress erythropoiesis. Oncogene. 2001; 20(28):3651–3664. doi: 10.1038/sj.onc.1204494 PMID: 11439328

78. Le A, Cooper CR, Gouw AM, Dinavahi R, Maitra A, Deck LM, et al. Inhibition of lactate dehydrogenase A induces oxidative stress and inhibits tumor progression. Proc Natl Acad Sci U S A. 2010; 107 (5):2037–42. doi: 10.1073/pnas.0914433107 PMID: 20133848

79. Wang YH, Israelsen WJ, Lee D, Yu VW, Jeanson NT, Clish CB, et al. Cell-state-specific metabolic dependency in hematopoiesis and leukemogenesis. Cell. 2014; 158(6):1309–23. doi: 10.1016/j.cell.2014.07.048 PMID: 25215489

80. Crauste F, Pujo-Menjouet L, Genieys S, Molina C, Gandrillon O. Adding Self-Renewal in Committed Erythroid Progenitors Improves the Biological Relevance of a Mathematical Model of Erythropoiesis. J Theor Biol. 2008; 250:322–338. doi: 10.1016/j.jtbi.2007.09.041 PMID: 17997418

81. Sahu D, Zhao Z, Tsen F, Cheng CF, Park R, Situ AJ, et al. A potentially common peptide target in secreted heat shock protein-90alpha for hypoxia-inducible factor-1alpha-positive tumors. Mol Biol Cell. 2012; 23(4):602–13. doi: 10.1091/mbc.E11-06-0575 PMID: 22190738

82. Takubo K, Nagamatsu G, Kobayashi CI, Nakamura-Ishizu A, Kobayashi H, Ikeda E, et al. Regulation of glycolysis by Pdk functions as a metabolic checkpoint for cell cycle quiescence in hematopoietic stem cells. Cell Stem Cell. 2013; 12(1):49–61. doi: 10.1016/j.stem.2012.10.011 PMID: 23290136

83. Oburoglu L, Romano M, Taylor N, Kinet S. Metabolic regulation of hematopoietic stem cell commitment and erythroid differentiation. Curr Opin Hematol. 2016; 23(3):198–205. doi: 10.1097/MOH.0000000000000234 PMID: 26871253

84. Wang J, Mi JQ, Debernardi A, Vitte AL, Emadali A, Meyer JA, et al. A six gene expression signature defines aggressive subtypes and predicts outcome in childhood and adult acute lymphoblastic leukemia. Oncotarget. 2015; 6(18):16527–42. doi: 10.18632/oncotarget.4113 PMID: 26001296

85. Stegle O, Teichmann SA, Marioni JC. Computational and analytical challenges in single-cell transcriptomics. Nat Rev Genet. 2015; doi: 10.1038/nrg3833 PMID: 25628217

86. Huang S. Non-genetic heterogeneity of cells in development: more than just noise. Development. 2009; 136(23):3853–62. doi: 10.1242/dev.035139 PMID: 19906852

87. Huang S, Guo YP, May G, Enver T. Bifurcation dynamics in lineage-commitment in bipotent progenitor cells. Dev Biol. 2007; 305(2):695–713. doi: 10.1016/j.ydbio.2007.02.036 PMID: 17412320

88. Mojtahedi M, Skupin A, Zhou J, Castaño IG, Leong-Quong RYY, Chang H, et al. Cell fate-decision as high-dimensional critical state transition. PLoS Biol. 2016; 14(12):e2000640. doi: 10.1371/journal.pbio.2000640

89. Chen P, Liu R, Chen L, Aihara K. Identifying critical differentiation state of MCF-7 cells for breast cancer by dynamical network biomarkers. Front Genet. 2015; 6:252. doi: 10.3389/fgene.2015.00252 PMID: 26284108

90. Stern S, Dror T, Stolovicki E, Brenner N, Braun E. Genome-wide transcriptional plasticity underlies cellular adaptation to novel challenge. Mol Syst Biol. 2007; 3:106. doi: 10.1038/msb4100147 PMID: 17453047

91. Paldi A. What makes the cell differentiate? Prog Biophys Mol Biol. 2012; 110(1):41–3. doi: 10.1016/j.pbiomolbio.2012.04.003 PMID: 22543273

92. Pelaez N, Gavalda-Miralles A, Wang B, Navarro HT, Gudjonson H, Rebay I, et al. Dynamics and heterogeneity of a fate determinant during transition towards cell differentiation. Elife. 2015; 4. doi: 10.7554/eLife.08924 PMID: 26583752

93. Kumar RM, Cahan P, Shalek AK, Satija R, DaleyKeyser AJ, Li H, et al. Deconstructing transcriptional heterogeneity in pluripotent stem cells. Nature. 2014; 516(7529):56–61. doi: 10.1038/nature13920 PMID: 25471879

Chapitre 5

Exploitation de l'aspect temporel : une approche itérative

Dans le chapitre 2, nous avons développé deux stratégies permettant de voir l'inférence de réseaux de régulation comme un problème statistique basé sur une pseudo-vraisemblance (approximation de Hartree) ou une véritable vraisemblance (auto-modèle Gamma-Binomial) : ces stratégies seront respectivement mises en pratique dans les chapitres 6 et 7. L'idée de base était de considérer les données de cellules uniques comme des échantillons indépendants de la loi stationnaire du modèle PDMP défini au chapitre 1. Cependant, comme nous l'avons vu au chapitre précédent, les données dont nous disposons ne correspondent pas à proprement parler à la loi stationnaire du PDMP mais plutôt à sa loi transitoire, évoluant dans le temps à partir d'une loi initiale correspondant au régime stationnaire pour une certaine valeur de θ, vers la loi stationnaire associée à une nouvelle valeur de θ. Intuitivement, la différence entre ces deux valeurs de θ est une matrice diagonale qui contient le *stimulus de différenciation*, c'est-à-dire la perturbation associée au changement de milieu à $t = 0$.

Dans ce chapitre, nous considérons une troisième stratégie dont l'objectif est précisément d'exploiter au maximum l'aspect temporel des données. Le prix à payer est que l'algorithme d'inférence, baptisé WASABI (*WAveS Analysis Based Inference*) et présenté dans l'article qui suit, est de type *brute-force* : il s'agit d'utiliser le modèle PDMP comme boite noire pour simuler des cellules correspondant à un réseau fixé, puis de comparer ces données *in silico* avec les vraies données. Le tout se fait en partant du graphe nul et en ajoutant des interactions de manière itérative, en utilisant de manière cruciale l'ordre dans lequel l'expression des gènes est visiblement perturbée par le stimulus. L'algorithme exploite le calcul parallèle en testant plusieurs réseaux candidats en même temps, et ce de manière branchante : le nombre de réseaux augmente donc au début, puis diminue progressivement au fur et à mesure que les branches sont élaguées, l'ajout d'interactions empêchant certains réseaux candidats de reproduire correctement les données.

En contrepartie, la méthode est extrêmement flexible. Il est par exemple possible de relâcher l'hypothèse que les taux de dégradation $d_{0,i}$ et $d_{1,i}$ sont constants (le PDMP n'est alors plus homogène en temps), ce qui n'est pas un luxe puisque les mesures des $d_{0,i}$ dont on dispose varient clairement au cours de l'expérience. Noter que dans cette situation, on n'utilise plus l'algorithme de simulation exact évoqué au chapitre 1, mais plutôt un schéma d'Euler hybride (détaillé dans l'article du chapitre 6) qui reste très satisfaisant en pratique.

Après environ deux semaines de calcul [1] à partir des données décrites au chapitre précédent, l'algorithme a renvoyé un certain nombre de réseaux candidats dont le meilleur – celui pour lequel le modèle PDMP génère des données dont les lois marginales correspondent le mieux aux observations – est montré sur la figure 6.B de l'article. De notre point de vue, il s'agit du réseau le plus abouti présenté dans cette thèse. Ce résultat permet de faire deux observations fondamentales :

- Il est impossible de reproduire la cinétique observée avec des voies de signalisation trop longues, car les demi-vies [2] des ARNm et des protéines sont tels que chaque intermédiaire ajoute un délai de réponse non négligeable. L'avantage de l'approche temporelle est qu'il a été possible de confronter de manière quantitative le temps de réponse de chaque réseau candidat avec les données réelles.
- De nombreux gènes sont directement affectés par le stimulus de différenciation. En outre, on constate l'absence de véritables *hubs* – des gènes qui servent de relai à un grand nombre d'autres – contrairement à ce que laisse parfois penser la littérature sur le sujet (Barabási et Oltvai, 2004).

Enfin, on remarque qu'un nombre important de boucles d'auto-activation ont été inférées, correspondant à la forme bimodale des distributions (cf. Figure 2). Ces boucles peuvent en fait représenter soit des activations directes des gènes sur eux mêmes, soit des boucles positives indirectes faisant intervenir des gènes non observés. Fonctionnellement, il pourrait s'agir d'une façon de mieux relayer l'information d'un gène à l'autre, ce qui se vérifie en tout cas très bien sur les simulations du modèle PDMP.

1. Effectuées sur les serveurs du centre de calcul de l'IN2P3 (CC-IN2P3, USR 6402) à Villeurbanne.
2. Par définition, la demi-vie d'une molécule est $\ln(2)/d$ où d est le taux de dégradation de la molécule.

WASABI: a dynamic iterative framework for gene regulatory network inference

Arnaud Bonnaffoux[1,2,3*], Ulysse Herbach[1,2,4], Angélique Richard[1], Anissa Guillemin[1], Sandrine Giraud[1], Pierre-Alexis Gros[3*], Olivier Gandrillon[1,2*]

1 Univ Lyon, ENS de Lyon, Univ Claude Bernard, CNRS UMR 5239, INSERM U1210, Laboratory of Biology and Modelling of the Cell, Lyon, France
2 Inria Team Dracula, Inria Center Grenoble Rhône-Alpes, Lyon, France
3 Cosmotech, Lyon, France
4 Univ Lyon, Université Claude Bernard Lyon 1, CNRS UMR 5208, Institut Camille Jordan, Villeurbanne, France

* arnaud.bonnaffoux@gmail.com
* pierre-alexis.gros@cosmotech.com
* olivier.gandrillon@ens-lyon.fr

Abstract

Inference of gene regulatory networks from gene expression data has been a long-standing and notoriously difficult task in systems biology. Recently, single-cell transcriptomic data have been massively used for gene regulatory network inference, with both successes and limitations. In the present work we propose an iterative algorithm called WASABI, dedicated to inferring a causal dynamical network from time-stamped single-cell data, which tackles some of the limitations associated with current approaches. We first introduce the concept of waves, which posits that the information provided by an external stimulus will affect genes one-by-one through a cascade, like waves spreading through a network. This concept allows us to infer the network one gene at a time, after genes have been ordered regarding their time of regulation. We then demonstrate the ability of WASABI to correctly infer small networks, which have been simulated *in silico* using a mechanistic model consisting of coupled piecewise-deterministic Markov processes for the proper description of gene expression at the single-cell level. We finally apply WASABI on *in vitro* generated data on an avian model of erythroid differentiation. The structure of the resulting gene regulatory network sheds a fascinating new light on the molecular mechanisms controlling this process. In particular, we find no evidence for hub genes and a much more distributed network structure than expected. Interestingly, we find that a majority of genes are under the direct control of the differentiation-inducing stimulus. In conclusion, WASABI is a versatile algorithm which should help biologists to fully exploit the power of time-stamped single-cell data.

Author summary

All cells have to make everyday decisions regarding their behavior in response to changing environment. Such decisions result from the dynamical behavior of an underlying gene regulatory network. Inferring the structure of such networks is an inverse problem which has occupied the systems biology community for decades. We propose in the present work a divide-and-conquer strategy called WASABI, which splits

the potentially untractable global problem into much simpler subproblems. We show that by adding one gene at a time, we can infer small networks, the behavior of which has been simulated *in silico* using a mechanistic model which incorporates the fundamentally probabilistic nature of the gene expression process. When applied to real-life data, our algorithm sheds a new fascinating light onto the molecular control of a differentiation process. It is our hope that WASABI will prove useful in helping biologists to fully exploit the power of time-stamped single-cell data.

Introduction

It is widely accepted that the process of cell decision making results from the behavior of an underlying dynamic gene regulatory network (GRN) [1]. The GRN maintains a stable state but can also respond to external perturbations to rearrange the gene expression pattern in a new relevant stable state, such as during a differentiation process. Its identification has raised great expectations for practical applications in network medicine [2] like somatic cells [3–5] or cancer cells reprogramming [6, 7]. The inference of such GRNs has, however, been a long-standing and notoriously difficult task in systems biology.

GRN inference was first based upon bulk data [8] using transcriptomics acquired through micro array or RNA sequencing (RNAseq) on populations of cells. Different strategies has been used for network inference including dynamic Bayesian networks [9, 10], boolean networks [11–13] and ordinary differential equations (ODE) [14] which can be coupled to Bayesian networks [15].

More recently, single-cell transcriptomic data, especially RNAseq [16], have been massively used for GRN inference (see [17, 18] for recent reviews). The arrival of those single-cell techniques led to question the fundamental limitations in the use of bulk data. Observations at the single-cell level demonstrated that any and every cell population is very heterogeneous [19–21]. Two different interpretations of the reasons behind single-cell heterogeneity led to two different research directions:

1. In the first view, this heterogeneity is nothing but a noise that blurs a fundamentally deterministic smooth process. This noise can have different origins, like technical noise ("dropouts") or temporal desynchronization as during a differentiation process. This view led to the re-use of the previous strategies and was at the basis of the reconstruction of a "pseudo-time" trajectory (reviewed in [22]). For example, SingleCellNet [23] and BoolTraineR [24] are based on boolean networks with preprocessing for cell clustering or pseudo-time reconstruction. Such asynchronous Boolean network models have been successfully applied in [25]. Other probabilistic algorithms such as SCOUP [26], SCIMITAR [27] or AR1MA1-VBEM [28] also use pseudo-time reconstruction complemented with correlation analysis. ODE based methods can be exemplified with SCODE [29] and InferenceSnapshot [30] algorithms which also use pseudo-time reconstruction.

2. The other view is based upon a representation of cells as dynamical systems [31, 32]. Within such a frame of mind, "noise" can be seen as the manifestation of the underlying molecular network itself. Therefore cell-to-cell variability is supposed to contain very valuable information regarding the gene expression process [33]. This view was advocated among others by [34], suggesting that heterogeneity is rooted into gene expression stochasticity, and that cell state dynamic is a highly stochastic process due to bursting that jumps discontinuously between micro-states. Dynamic algorithms like SINCERITIES [35] are based upon comparison of gene expression distributions, incorporating (although not explicitly) the bursty nature of gene expression. We have recently described a more explicit network formulation view based upon the coupling of probabilistic two-state models of gene expression [36]. We devised a statistical hidden

Markov model with interpretable parameters, which was shown to correctly infer small two-gene networks [36].

Despite their contributions and successes, all existing GRN inference approaches are confronted to some limitations:

1. The inference of interactions through the calculation of correlation between gene expression, whether based upon or linear [27] or non-linear [26] assumptions, is problematic. Such correlations can only reproduce events that have been previously observed. As a consequence, predictions of GRN response to new stimulus or modifications is not possible. Furthermore, correlation should not be mistaken for causality. The absence of causal relationship severely hampers any predictive ability of the inferred GRN.

2. The very possibility of making predictions relies upon our ability to simulate the behavior of candidate networks. This implicitly implies that network topologies are explicitly defined. Nevertheless, several inference algorithms [27–29, 35] propose a set of possible interactions with independent confidence levels, generally represented by an interaction matrix. The number of possible actionable networks deduced from combining such interactions is often too large to be simulated.

3. Regulatory proteins within a GRN are usually restricted to transcription factors (TF), like in [24, 26–30]. Possible indirect interactions are completely ignored. A trivial example is a gene encoding a protein that induces the nuclear translocation of a constitutive TF. In this case, the regulator gene will indirectly regulate TF target genes, and its effect will be crucial in understanding the GRN behavior.

4. Most single-cell inference algorithms rely upon the use of a single type of data, namely transcriptomics. By doing so, they implicitly assume protein levels to be positively correlated with RNA amounts, which has been proven to be wrong in case of post-translational regulation (see [33] for an illustration in circadian clock). Besides, at single-cell scale, mRNA and proteins typically have a poor linear correlation [34], even in the absence of post-translational regulation.

5. The choices of biological assumptions are also important for the biological relevance of GRN models. The use of statistical tools can be really powerful to handle large-scale network inference problem with thousand of genes, but the price to pay is loss of biological representativeness. By definition a model is a simplification of the system, but when simplifying assumptions are induced by mathematical tools, like linear [27–29, 35] or binary (boolean) requirements [23, 24], the model becomes solvable at the expense of its biological relevance.

In the present work we address the above limitations and we propose an iterative algorithm called WASABI, dedicated to inferring a causal dynamical network from time-stamped single-cell transcriptomic data, with the capability to integrate protein measurements. In the first part we present the WASABI framework which is based upon a mechanistic model for gene-gene interactions [36]. In the second part we benchmark our algorithm using *in silico* GRNs with realistic gene parameter values. Finally we apply WASABI on our *in vitro* data [37] and analyze the resulting GRN candidates.

Results

Our goal is to infer causalities involved in GRN through analysis of dynamic multi-scale/level data with the help of a mechanistic model [36]. We first present an overview of the WASABI principles and framework. We then benchmark its ability to correctly infer *in silico*-generated toy GRNs. Finally, we apply WASABI on our *in vitro* data on avian erythroid differentiation model [38] to generate biologically relevant GRN candidates.

WASABI inference principles and implementation

WASABI is a framework built on a novel inference strategy based on the concept of "waves". We posit that the information provided by an external stimulus will affect genes one-by-one through a cascade, like waves spreading through a network (Fig 1-A). This wave process harbors an inertia determined by mRNA and protein half-lives which are given by their degradation rate.

By definition, causality is the link between cause and consequence, and causes always precede consequences. This temporal property is therefore of paramount importance for causality inference using dynamic data. In our mechanistic and stochastic model of GRN [36] (detailed in Method section Fig 7), the cause corresponds either to the protein of the regulating gene or a stimulus, which level modulates as a consequence the promoter state switching rates k_{on} (i.e. probability to switch from inactive to active state) and k_{off} (active to inactive) of the target gene. A direct consequence of causality principle for GRNs is that a dynamical change in promoter activity can only be due to a previous perturbation of a regulating protein or stimulus. For example, assuming that the system starts at a steady-state, early activated genes (referred to as early genes) can only be regulated by the stimulus, because it is the only possible cause for their initial evolution. An illustration is given in Fig 1-A: gene A initial variation can only be due to the stimulus and not by the feedback from gene C, which will occur later. A generalization of these concepts is that for a given time after the stimulus, we can infer the subnetwork composed exclusively by genes affected by the spreading of information up to this time. Therefore we can infer iteratively the network by adding one gene at a time (Fig 1-D) regarding their promoter wave time order (Fig 1-B) and comparing with protein wave time of previous added genes (Fig 1-C).

For this, we need to estimate promoter and protein wave times for each gene and then sort them by promoter wave time. We define the promoter activity level by the $k_{\text{on}}/(k_{\text{on}} + k_{\text{off}})$ ratio, which corresponds to the local mean active duration (Fig 1-B). Promoter wave time is defined as the inflection time point of promoter activity level where 50% of evolution between minimum and maximum is reached. Since promoter activity is not observable, we estimate the inflection time point of mean RNA level from single-cell transcriptomic kinetic data [37], and retrieve the delay induced by RNA degradation to deduce promoter wave time. Protein wave times correspond to the inflection point of mean protein level, which can be directly observed with our proteomic data [39]. A detailed description of promoter and protein wave time estimation can be found in the Method section. One should note that a gene can have more than one wave time in case of non monotonous variation of promoter activity, due to feedbacks (like gene A in our example) or incoherent feed-forward loop.

The WASABI inference process (Fig 1-C) takes advantage of the gene wave time sorting by adopting a divide and conquer strategy. We remind that a main assumption of our interaction model is the separation between mRNA and protein timescales [36]. As a consequence, for a given interaction between a regulator gene and a regulated gene, the regulated promoter wave time should be compatible with the regulator protein wave time. At each step, WASABI proposes a list of possible regulators in order to reduce the dimension of the inference problem. This list is limited to regulators with compatible protein wave time within the range of 30 hours before and 20 hours after the promoter wave time of the added regulated gene. This constraint has been set up from *in silico* study (see next section). For example, in Fig 1, gene B can be regulated by gene A or D since their protein wave time are close to gene B promoter wave time. Gene C can be regulated by gene B or D, but not A because its protein wave time is too earlier compared to gene C promoter wave time.

For new proposed interactions, a typical calibration algorithm can be used to finely tune interaction parameter in order to fit simulated mRNA marginal distribution with

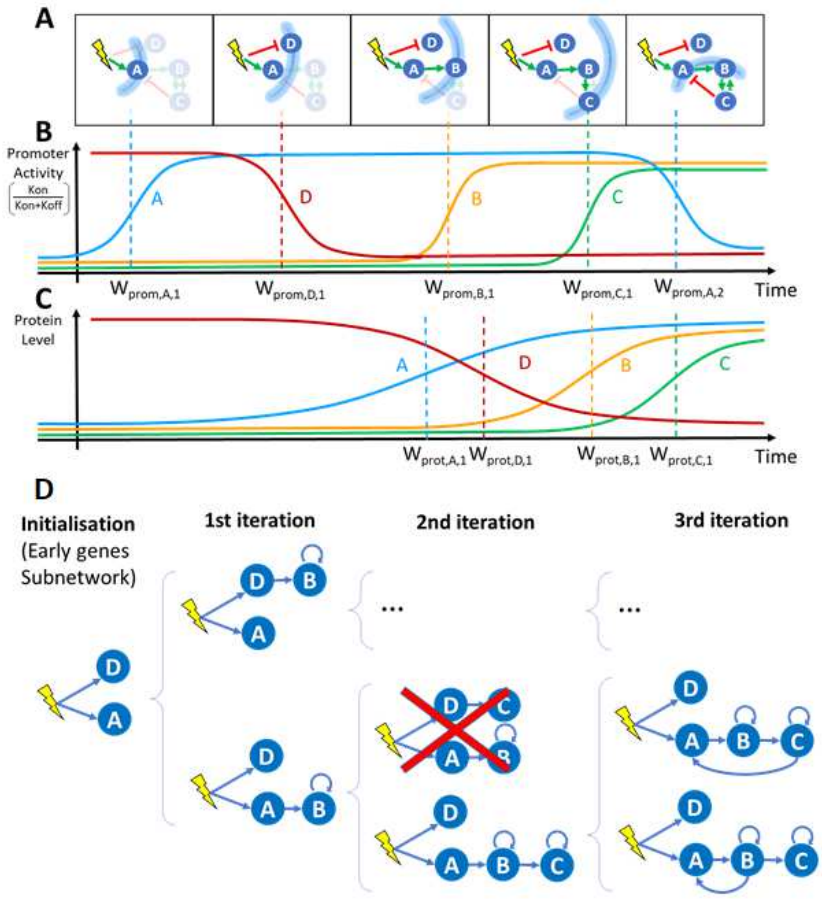


Fig 1. WASABI at a glance. A) Schematic view of a GRN: the stimulus is represented by a yellow flash, genes by blue circles and interactions by green (activation) or red (inhibition) arrows. The stimulus-induced information propagation is represented by blue arcs corresponding to wave times. Genes and interactions that are not affected by information at a given wave time are shaded. At wave time 5, gene C returns information on gene A and B by feedback interaction creating a backflow wave. B) Promoter wave times: Promoter wave times correspond to inflections point of gene promoter activity defined as the $k_{on}/(k_{on}+k_{off})$ ratio. C) Protein wave times: Protein wave times correspond to inflections point of mean protein level. D) Inference process. Blue arrows represent interactions selected for calibration. Based on promoter waves classification genes are iteratively added to sub-GRN previously inferred to get new expanded GRN. Calibration is performed by comparison of marginal RNA distributions between *in silico* and *in vitro* data. Inference is initialized with calibration of early genes interaction with stimulus, which gives initial sub-GRN. Latter genes are added one by one to a subset of potential regulators for which a protein wave time is close enough to the added gene promoter wave time. Each resulting sub-GRN is selected regarding its fit distance to *in vitro* data. If fit distance is too important sub-GRN can be eliminated (red cross). An important benefit of this process is the possibility to parallelize the sub-GRN calibrations over several cores, which results in a linear computational time regarding the number of genes. Note that only a fraction of all tested sub-GRN is shown.

experimental marginal distribution from transcriptomic single-cell data. To avoid over-fitting issues, only efficiency interaction parameter $\theta_{i,j}$ (Fig 7) is tuned. To estimate fitting quality we define a GRN fit distance based on the Kantorovitch distances between simulated and experimental mRNA marginal distributions (please refer to Method section for a detailed description of interaction function and calibration process). If the resulting fitting is judged unsatisfactory (i.e. GRN fit distance is greater than a threshold), the sub-GRN candidate is pruned. For genes presenting several waves, like gene A, each wave will be separately inferred. For example, gene A initial increase is fitted during initialization step, but only the first experimental time points during promoter activity increase will be used for calibration. Genes B and C regulated after gene A up-regulation will be added to expand sub-GRN candidates. Finally, the wave corresponding to gene A down-regulation is then fitted considering possible interactions with previously added genes (namely gene B and C), which permits the creation of feedback loops or incoherent feed-forward loops.

Positive feedback loops cannot be easily detected by wave analysis because they only accelerate, and eventually amplify, gene expression. Yet, their inference is important for the GRN behavior since they create a dynamic memory and, for example, may thus participate to irreversibility of the differentiation process. To this end, we developed an algorithm to detect the effect of positive feedback loops on gene distribution before the iterative inference (see Supporting information). We modeled the effect of positive feedback loops by adding auto-positive interactions. Note that such a loop does not necessarily mean that the protein directly activates its own promoter: it simply means that the gene is influenced by a positive feedback, which can be of different nature. For example, in the GRN presented in Fig 1-A, genes B and C mutually create a positive feedback loop. If this positive feedback loop is detected we consider that each gene has its own auto-positive interaction as illustrated in Fig 1-C. Positive feedback loops could also arise from the existence of self-reinforcing open chromatin states [40] or be due to the fact that binding of one TF can shape the DNA in a manner that it promotes the binding of the second TF [41].

In silico benchmarking

We decided to first calibrate and then assess WASABI performance in a controlled and representative setting.

Calibration of inference parameters

In the first phase we assessed some critical values to be used in the inference process. We generate realistic GRNs (Fig2-A) where 20 genes from *in vitro* data were randomly selected with associated *in vitro* estimated parameters (see Supporting information). Interactions were randomly defined in order to create cascade networks with no feedback nor auto-positive feedback as an initial assessment phase.

We limited ourselves to 4 network levels (with 5 genes at each level, see Fig2-A for an example) because we observed that the information provided by the stimulus is almost completely lost after 4 successive interactions in the absence of positive feedback loops. This is very likely caused by the fact that each gene level adds both some intrinsic noise, due to the bursty nature of gene expression, as well as a filtering attenuation effect due to RNA and protein degradation.

We first analyzed the special case of early genes that are directly regulated by the stimulus (Fig2-B). Their promoter wave times were lower than all other genes but one. Therefore we can identify early genes with good confidence, based on comparison of their promoter wave time with a threshold. Given these *in silico* results, we then decided in the WASABI pre-processing step to assume that genes with a promoter wave

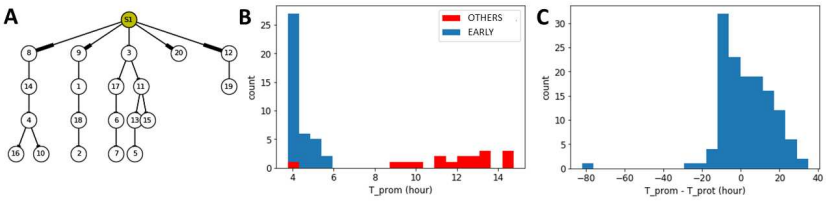


Fig 2. Cascade *in silico* GRN A) Cascade GRN types are generated to study wave dynamics. Genes correspond to *in vitro* ones with their estimated parameters. S1 corresponds to stimulus. Genes are identified by our list gene ID. B) Based on 10 *in silico* GRN we compare promoter wave time of early genes (blue) with other genes (red). Displayed are promoter waves with a wave time lower than 15h for graph clarity. C) For each interactions of 10 *in silico* GRNs we compute the difference between estimated regulated promoter wave time minus its regulator protein wave time. Distribution of promoter/protein wave time difference is given for all interactions of all *in silico* GRNs.

time below 5h must be early genes, and that genes with a promoter wave time larger than 7h can not be early genes. Interactions between the stimulus and intermediate genes, with promoter wave times between 5h and 7h, have to be tested during the inference iterative process and preserved or not.

We then assessed what would be the acceptable bounds for the difference between regulator protein wave time and regulated gene promoter activity. 10 *in silico* cascade GRNs were generated and simulated for 500 cells to generate population data from which both protein and promoter wave times were estimated for each gene. Based on these data, we computed the difference between estimated regulated promoter wave time minus its regulator protein wave time for all interactions in all networks. The distribution of these wave differences is given in Fig2-C. One can notice that some wave differences had negative values. This is due to the shape of the Hill interaction function (see eq3 in Method section) with a moderate transition slope ($\gamma = 2$). If the protein threshold (which corresponds to typical EC50 value) is too close to the initial protein level, then a slight protein increase will activate target promoter activity. Therefore, promoter activity will be saturated before regulator protein level and thus the difference of associated wave times is negative. This shows that one can accelerate or delay information, depending on the protein threshold value. In order to be conservative during the inference process, we set the RNA/Protein wave difference bounds to [−20h; 30h] in accordance with the distribution in Fig2-C. One should note that this range, even if conservative, already removes two thirds of all possible interactions, thereby reducing the inference complexity.

We finally observed that for interactions with genes harboring an auto-positive feedback, wave time differences could be larger. In this case, wave difference bounds were estimated to [−30h, 50h] (see supporting information). We interpret this enlargement by an under-sampling time resolution problem since auto-positive feedback results in a sharper transition. As a consequence, promoter state transition from inactive to active is much faster: if it happens between two experimental time points, we cannot detect precisely its wave time.

Inference of *in silico* GRNs

WASABI was then tested for its ability to infer *in silico* GRNs (complete definition in supporting information) from which we previously simulated experimental data for mRNA and protein levels at single-cell and population scales. We first assessed the simplest scenario with a toy GRN composed of two branches with no feedback (a

cascade GRN; Fig 3-A). The GRN was limited to 6 genes and to 3 levels in order to reduce computational constraints. Nevertheless, even in such a simple case, the inference problem is already a highly complex challenge with more than 10^{20} possible directed networks.

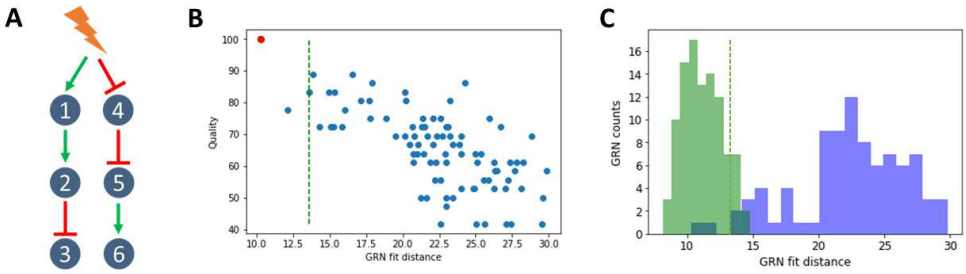


Fig 3. *In silico* cascade GRN inference A) The cascade GRN. Genes parameters were taken from *in vitro* estimations to mimic realistic behavior. Experimental data were generated to obtain time courses of transciptomic data, at single-cell and population scale, and also proteomic data at population scale. B) WASABI was run to infer *in silico* cascade GRN and generated 88 candidates. A dot represents a network candidate with its associated fit distance and inference quality (percentage of true interactions). True GRN is inferred (red dot, 100% quality). Acceptable maximum fit distance (green dashed line) corresponds to variability of true GRN fit distance. Its computation is detailed in figure C. 3 GRN candidates (including the true one) have a fit distance below threshold. C) Variability of true GRN fit distance (green dashed line in figures B and C) is estimated as the threshold where 95% of true GRN fit distance is below. Fit distance distribution is represented for true GRN (green) and candidates (blue) for cascade *in silico* GRN benchmark. True GRNs are calibrated by WASABI directed inference while candidates are inferred from non-directed inference. Fit distance represents similitude between candidates generated data and reference experimental data.

Wave times were estimated for each gene from simulated population data for RNA and protein (data available in supporting information). Table 1 provides estimated waves time for the cascade GRN. It is clear that the gene network level is correctly reproduced by wave times.

Table 1. Wave times. Promoter and protein wave times (in hours) estimated from *in silico* simulated data.

GRN	Gene	$W_{promoter}$	$W_{protein}$
Cascade	4	4.12	12.99
	1	4.26	22.33
	5	15.19	45.50
	2	17.67	44.88
	3	37.88	60.10
	6	40.06	60.72

We then ran WASABI on the generated data and obtained 88 GRN candidates (Fig 3-B). The huge reduction in numbers (from 10^{20} to 88) illustrates the power of WASABI to reduce complexity by applying our waves-based constraints. We defined two measures for further assessing the relevance of our candidates:

1. *Quality* quantifies proportion of real interactions that are conserved in the candidate network (see supporting information for a detailed description). A 100% corresponds to the true GRN.

2. A *fit distance*, defined as the mean of the 3 worst gene fit distances, where gene fit distance is the mean of the 3 worst Kantorovitch distances [42] among time points (see the Methods section).

We observed a clear trend that higher quality is associated with a lower fit distance (Fig 3-B), which we denote as a good specificity. When inferring *in vitro* GRNs, one does not have access to quality score, contrary to fit distance. Hence, having a good specificity enables to confidently estimate the quality of GRN candidates from their fit distance. Thus, this result demonstrates that our fit distance criterion can be used for GRN inference. Nevertheless, even in the case of a purely *in silico* approach, quality and fit distance can not be linked by a linear relationship. In other words, the best fit distance can not be taken for the best quality (see below for other toy GRNs). This is likely to be due to both the stochastic gene expression process as well as the estimation procedure. We therefore needed to estimate an acceptable maximum fit distance threshold for true GRN. For this, we ran directed inferences, where WASABI was informed beforehand of the true interactions, but calibration was still run to calibrate interaction parameters. We ran 100 directed inferences and defined the maximum acceptable fit distance (Fig 3-C) as the distance for which 95% of true GRN fit distance was below. This threshold could also be used as a pruning threshold (green dashed line in Fig 3-B) in subsequent iterative inferences, thereby progressively reducing the number of acceptable candidates. We then analyzed a situation where we added either an auto-activation loop or a negative feedback (Fig 4-A and C and supporting information for estimated wave times).

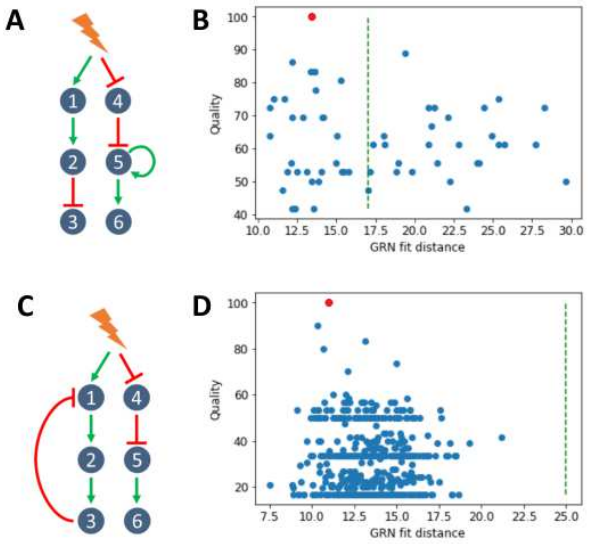


Fig 4. *In silico* GRN with feedbacks A) Addition of one positive feedback onto the cascade GRN. B) WASABI was run to infer *in silico* cascade GRN with a positive feedback and generated 59 candidates, 31 of which having an acceptable fit distance. See legend to Fig 3-B for details. C) Addition of one negative feedback onto the cascade GRN. D) WASABI was run to infer *in silico* cascade GRN with a negative feedback and generated 476 candidates, all of which having an acceptable fit distance. See legend to Fig 3-B for details.

In both cases, GRN inference specificity was lower than for cascade network inference.

Nevertheless in both cases the true network was inferred and ranked among the first candidates regarding their fit distance (Fig 4-B and D), demonstrating that WASABI is able to infer auto-positive and negative feedback patterns. However there were more candidates below the acceptable maximum fit distance threshold and there was no obvious correlation between high quality and low fit distance. We think it could be due to data under-sampling regarding the network dynamics (see upper and discussion).

In vitro application of WASABI

We then applied WASABI on our *in vitro* data, which consists in time stamped single-cell transcriptomic [37] and bulk proteomic data [39] acquired during T2EC differentiation [38], to propose relevant GRN candidates.

We first estimated the wave times (Fig 5). Promoter waves ranged from very early genes regulated before 1h to late genes regulated after 60h. Promoter activity appeared bimodal with an important group of genes regulated before 20h and a second group after 30h. Protein wave distribution was more uniform from 10h to 60h, in accordance with a slower dynamics for proteins. Remarkably, 10 genes harbored non-monotonous evolution of their promoter activity with a transient increase. It can be explained by the presence of a negative feedback loop or an incoherent feed-forward interaction. These results demonstrate that real *in vitro* GRN exhibits distinguishable "waves".

In order to limit computation time, we decided to further restrict the inference to the most important genes in term of the dynamical behavior of the GRN. We first detected 25 genes that are defined as early with a promoter time lower than 5h. We then defined a second class of genes called "readout" which are influenced by the network state but can not influence in return other genes. Their role for final cell state is certainly crucial, but their influence on the GRN behavior is nevertheless limited. 41 genes were classified as readout so that 24 genes were kept for iterative inference, in addition to the 25 early genes. 9 of these 24 genes have 2 waves due to transient increase, which means that we have 33 waves to iteratively infer.

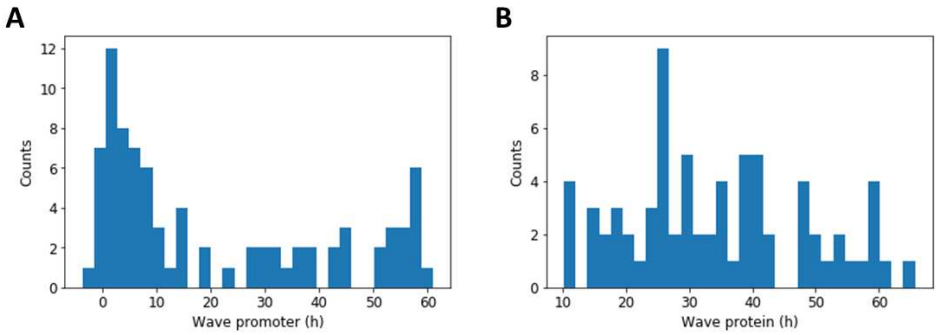


Fig 5. Promoter and protein wave time distributions. Distribution of *in vitro* promoter (A) and protein (B) wave times for all genes estimated from RNA and proteomic data at population scale. Counts represent number of genes. Note: a gene can have several waves for its promoter or protein.

In vitro GRN candidates

After running for 16 days using 400 computational cores, WASABI returned a list of 381 GRN candidates. Candidate fit distances showed a very homogeneous distribution (see supporting information) with a mean value around 30, together with outliers at much higher distances. Removing those outliers left us with 364 candidates. Compared to

inference of *in silico* GRN, *in vitro* fitting is less precise, as we could expect. But it is an appreciable performance and it demonstrates that our GRN model is relevant.

We then analyzed the extent of similarities among the GRN candidates regarding their topology by building a consensus interaction matrix (Fig6-A). The first observation is that the matrix is very sparse (except for early genes in first raw and auto-positive feedbacks in diagonal) meaning that a sparse network is sufficient for reproducing our *in vitro* data. We also clearly see that all candidate GRNs share closely related topologies. This is clearly obvious for early genes and auto-positive feedbacks. Columns with interaction rates lower than 100% correspond to latest integrated genes in the iterative inference process with gene index (from earlier to later) 70, 73, 89, 69 and 29. Results from existing algorithms are usually presented in such a form, where the percent of interactions are plotted [27–29, 35]. But one main advantage of our approach is that it actually proposes real GRN candidates, which may be individually examined.

We therefore took a closer look at the "best" candidate network, with the lowest Fit distance to the data (Fig6-B). We observed very interesting and somewhat unexpected patterns:

1. Most of the genes (84%) with an auto-activation loop. As mentioned earlier, this was a consensual finding among the candidate networks. It is striking because typical GRN graphs found in the literature do not have such predominance of auto-positive feedbacks.

2. A very large number of genes were found to be early genes that are under the direct control of the stimulus. It is noticeable that most of them were found to be inhibited by the stimulus, and to control not more than one other gene at one next level.

3. We previously described the genes whose product participates in the sterol synthesis pathway, as being enriched for early genes [37]. This was confirmed by our network analysis, with only one sterol-related gene not being an early gene.

4. Among 7 early genes that are positively controlled by the stimulus, 6 are influenced by an incoherent feedforward loop, certainly to reproduce their transient increase experimentally observed [37].

5. One important general rule is that the network depth is limited to 3 genes. One should note that this is not imposed by WASABI which can create networks with unlimited depth. It is consistent with our analysis on signal propagation properties in *in silico* GRN. If network depth is too large, signal is too damped and delayed to accurately reproduce experimental data.

6. One do not see network hubs in the classical sense. The genes in the GRNs are connected to at most four neighbors. The most impacting "node" is the stimulus itself.

7. One can also observe that the more one progress within the network, the less consensual the interaction are. Adding the leaves in the inference process might help to stabilize those late interactions.

Altogether those results show the power of WASABI to offer a brand-new vision of the dynamical control of differentiation.

Discussion

In the present work we introduced WASABI as a new iterative approach for GRN inference based on single-cell data. We benchmarked it on a representative *in silico* environment before its application on *in vitro* data.

WASABI tackles GRN inference limitations

We are convinced that WASABI has the ability to tackle some general GRN inference issues.

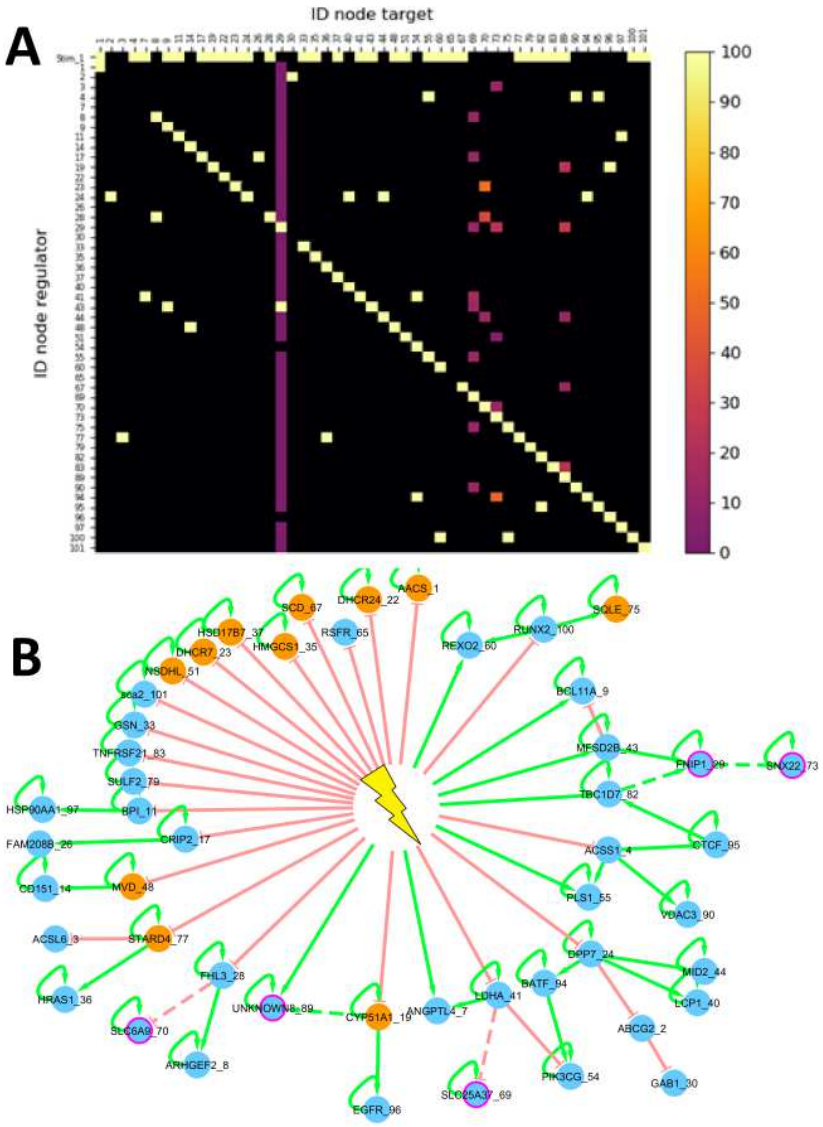


Fig 6. Inference from *in vitro* data A) *In vitro* interaction consensus matrix. Each square in the matrix represents either the absence of any interaction, in black, or the presence of an interaction, the frequency of which is color-coded, between the considered regulator ID (row) and regulated gene ID (column). First row correspond to stimulus interactions. B) Best candidate. Green: positive interaction; red: negative interaction; plain lines: interactions found in 100% of the candidates; dashed lines: interaction found only in some of the candidates; orange: genes the product of which participates to the sterol synthesis pathway; purple: 5 last added genes during iterative inference.

1. WASABI goes beyond mere correlations to infer causalities from time stamped data analysis as demonstrated on *in silico* benchmark (Fig3) even in the presence of circular causations (Fig4), based upon the principle that the cause precedes the effect.
2. Contrary to most GRN inference algorithms [27–29, 35] based upon the inference of interactions, WASABI is network centered and generates several candidates with explicitly defined networks topology (Fig6-B), which is required for prediction making and simulation capability. Generating a list of interactions and their frequency from

such candidates is a trivial task (Fig6-A) whereas the reverse is usually not possible. Moreover, WASABI explicitly integrates the presence of an external stimulus, which surprisingly is never modeled in other approaches based on single-cell data analysis. It could be very instrumental for simulating for example pulses of stimuli.

3. WASABI is not restricted to transcription factors (TFs). Most of the *in vitro* genes we modeled are not TFs. This is possible thanks to the use of our mechanistic model [36] which integrates the notion of timescale separation. It assumes that every biochemical reaction such as metabolic changes, nuclear translocations or post-translational modifications are faster than gene expression dynamics (imposed by mRNA and protein half-life) and that they can be abstracted in the interaction between 2 genes. Our interaction model is therefore an approximation of the underlying biochemical cascade reactions. This should be kept in mind when interpreting an interaction in our GRN: many intermediaries (fast) reactions may be hidden behind this interaction.

4. Optionally, WASABI offers the capability to integrate proteomic data to reproduce translational or post-translational regulation. Our proteomic data [39] demonstrate that nearly half of detected genes exhibit mRNA/protein uncoupling during differentiation and allowed to estimate the time evolution of protein production and degradation rates. Nevertheless, we are not fully explanatory since we do not infer causalities of these parameters evolution. This is a source of improvement discussed later.

5. We deliberately developed WASABI in a "brute force" computational way to guarantee its biological relevance and versatility. This allowed to minimize simplifying assumptions potentially necessary for mathematical formulations. During calibration, we used a simple Euler solver to simulate our networks within model (1). This facilitates addition of any new biological assumption, like post-translation regulations, without modifying the WASABI framework, making it very versatile. Thanks to the splitting and parallelization allowed by WASABI original gene-by-gene iterative inference process, the inference problem becomes linear regarding the network size, whereas typical GRN inference algorithms face combinatorial curse. This strategy also allowed the use of High Parallel Computing (HPC) which is a powerful tool that remains underused for GRN inference [23, 43].

WASABI performances, improvements and next steps

WASABI has been developed and tested on an *in silico* controlled environment before its application on *in vitro* data. Each *in silico* network true topology was successfully inferred. Cascade type GRN is perfectly inferred (Fig3) with an excellent specificity. Auto-positive and negative feedback networks (Fig4) were also inferred, demonstrating WASABI's ability to infer circular causations, but specificity is lower. This might be due to a time sampling of experimental data being longer than the network dynamic time scale. Auto-positive feedback creates a switch like response, the dynamic of which is much quicker than simple activation. Thus, to capture accurately auto-positive feedback wave time, we should use high frequency time sample for RNA experimental data during auto-positive feedback activation short period. For negative feedback interactions, WASABI calibrated initial increase considering only first experimental time points before feedback effect. Consequently, precision of first interaction was decreased and more false positive sub-GRN candidates were selected. Increasing the frequency of experimental time sampling during initial phase should overcome this problem.

As it stands our mechanistic model is only accounting for transcriptional regulation through proteins. It does not take into account other putative regulation level, including translational or post-translational regulations, or regulation of the mRNA half-life, although there is ample evidence that such regulation might be relevant [44, 45].

Provided that sufficient data is available, it would be straightforward to integrate such information within the WASABI framework. For example, the estimation of the degradation rates at the single-cell level for mRNAs and proteins has recently been described [46], the distribution of which could then be used as an input into the WASABI inference scheme.

Cooperativity and redundancies are not considered in the current WASABI framework, so that a gene can only be regulated by one gene, except for negative feedback or incoherent feedforward interactions. However, many experimentally curated GRN show evidence for cooperations (2 genes are needed to activate a third gene) or redundant interactions (2 genes independently activating a third gene) [47]. We intentionally did not considered such multi-interactions because our current calibration algorithm relies on the comparison of marginal distributions which are not sufficiently informative for inferring cooperative effects. It is our belief that the use of joint distribution of two genes or more should enable such inference. We previously developed in our group a GRN inference algorithm which is based on joint distribution analysis [36] but which does not consider time evolution. We are therefore planning to integrate joint-distribution-based analyses within the WASABI framework in order to improve calibration, by upgrading the objective function with measurement considering joint-distribution comparison.

HPC capacities used during iterative inference impacts WASABI accuracy. Indeed late iterations are supposed more discriminative than the first one because false GRN candidates have accumulated too many wrong interactions so that calibration is not able to compensate for errors. However, if the expansion phase is limited by available computational nodes, the true candidate may be eliminated because at this stage inference is not discriminative enough. Therefore improving computing performances would represent an important refinement and we have initiated preliminary studies in that direction [43].

Nevertheless, despite all possible improvements, GRN inference will remain *per se* an asymptotically solvable problem due to inferability limitations [48], intrinsic biological stochasticity, experimental noise and sampling. This is why we propose a set of GRN candidates with acceptable confidence level. A natural companion of the WASABI approach would be a phase of design of experiments (DOE) specifically aiming at selecting the most informative experiments to discriminate among the candidates. Such DOE procedures have already been developed for GRN inference, but none of them takes into account the mechanistic aspects and the stochasticity of gene expression [48, 49]. Extending the DOE framework to stochastic models is currently being developed in our group.

New insights on typical GRN topology

The application of WASABI on our *in vitro* model of differentiation generated several GRN candidates with a very interesting consensus topology (Fig6).

1. We can see that the stimulus (i.e. medium change [37]) is a central regulator of our GRN. We are strongly confident with this result because initial RNA kinetic of early genes can only be explained by fast regulation at promoter level several minutes after stimulation. Proteins dynamics are way too slow to justify these early variations.

2. 22 of the 29 inferred early genes are inhibited by the stimulus, while inhibitions are only present in 7 of the 28 non-early interactions. Thus inhibitions are overrepresented in stimulus-early genes interactions. An interpretation is that most of genes are auto-activated and their inhibition requires a strong and long enough signal to eliminate remaining auto-activated proteins. A constant and strong stimulus should be very efficient for this role like in [32] where stimulus long duration and high amplitude is required to overcome an auto-activation feedback effect. It could be very interesting in

that respect to assess how the network would respond to a temporary stimulus, mimicking the commitment experiment described in [37] or [50].

3. None of our GRN candidates do contain so-called "hubs genes" affecting in parallel many genes, whereas existing GRN inferred generally present consequent hubs [26, 28, 29, 35] . A possible interpretation is that hub identifications is mostly a by-product of correlation analysis. This interpretation is in line with the sparse nature of our candidate networks, as compared to some previous network (see e.g. [25] or [51]). This strongly departs with the assumption that small-world network might represent "universal laws" [52].

4. In order to reproduce non-monotonous gene expression variations, WASABI inferred systematically incoherent feedforward pattern instead of "simpler" negative feedback. This result is interesting because nothing in WASABI explain this bias since *in silico* benchmarking proved that WASABI is able to infer simple negative feedbacks (Fig4). Such "paradoxical components" have been proposed to provide robustness, generate temporal pulses, and provide fold-change detection [53].

5. WASABI candidates are limited in network depth by a maximum of 3 levels. We did not include readout genes during inference but addition of these genes would only increase GRN candidate depth by one level. GRN realistic candidates depth are thus limited by 4 levels. This might be due to the fact that information can only be relayed by limited number of intermediaries because of induced time delay, damping and noise. Indeed, general mechanism of molecules production/degradation behaves exactly as a low pass filter with a cutting frequency equivalent to the molecule degradation rate. Furthermore, protein information will be transmitted at the promoter target level by modulation of burst size and frequency, which are stochastic parameters, thereby adding noise to the original signal.

Such a strong limitation for information carrying capacity in GRN is at stake with long differentiation sequences, say from the hematopoietic stem cell to a fully committed cell. In such a case, tens of genes will have to be sequentially regulated. This might be resolved by the addition of auto-positive feedbacks. Such auto-positive feedbacks will create a dynamic memory whereby the information is maintained even in the absence of the initial information. An important implication is the loss of correlation between auto-activated gene and its regulator gene. Consequently, all algorithms based on stationary RNA single-cell correlation [26, 27] will hardly catch regulators of auto-activated genes.

Considering the importance of auto-positive feedback benefits on GRN information transfert, it is therefore not surprising to see that more than 80% of our GRN genes present auto-positive feedback signatures in their RNA distribution. Moreover, experimentally observed auto-positive feedback influence is stronger in our *in vitro* model than in our *in silico* models. Such a strong prevalence of auto-positive feedbacks has also been observed in a network underlying germ cell differentiation [51]. As mentioned earlier, care should be taken in interpreting such positive influences, which very likely rely on indirect influences, like epigenomic remodeling.

Materials and Methods

Mechanistic GRN model

Our approach is based on a mechanistic model that has been previously introduced in [36] and which is summed-up in Fig 7.

In all that follows, we consider a set of G interacting genes potentially influenced by a stimulus level Q. Each gene i is described by its promoter state $E_i = 0$ (off) or 1 (on), its mRNA level M_i and its protein level P_i. We recall the model definition in the

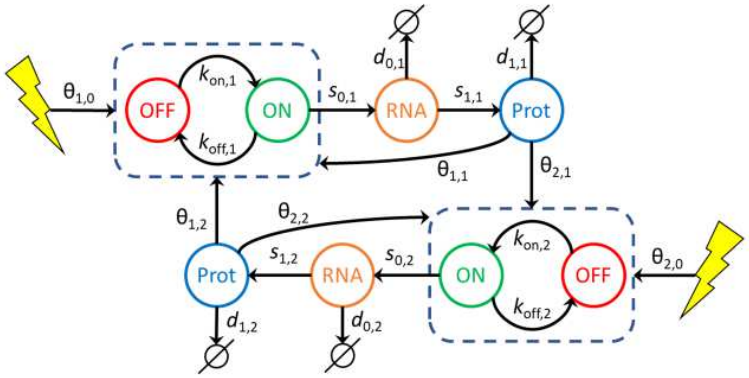


Fig 7. GRN mechanistic and stochastic model. Our GRN model is composed of coupled piecewise deterministic Markov processes. In this example 2 genes are coupled. A gene i is represented by its promoter state (dashed box) which can switch randomly from ON to OFF, and OFF to ON, respectively at $k_{\text{on},i}$ and $k_{\text{off},i}$ mean rate. When promoter state is ON, mRNA molecules are continuously produced at a $s_{0,i}$ rate. mRNA molecules are constantly degraded at a $d_{0,i}$ rate. Proteins are constantly translated from mRNA at a $s_{1,i}$ rate and degraded at a $d_{1,i}$ rate. The interaction between a regulator gene j and a target gene i is defined by the dependence of $k_{\text{on},i}$ and $k_{\text{off},i}$ with respect to the protein level P_j of gene j and the interaction parameter $\theta_{i,j}$. Likewise, a stimulus (yellow flash) can regulate a gene i by modulating its $k_{\text{on},i}$ and $k_{\text{off},i}$ switching rates with interaction parameter $\theta_{i,0}$.

following equation, together with notations that will be extensively used throughout this article.

$$\begin{cases} E_i(t) : 0 \xrightarrow{k_{\text{on}}} 1, \quad 1 \xrightarrow{k_{\text{off}}} 0 \\ M_i'(t) = s_{0,i}E_i(t) - d_{0,i}M_i(t) \\ P_i'(t) = s_{1,i}M_i(t) - d_{1,i}P_i(t) \end{cases} \tag{1}$$

The first line in model (1) represents a discrete, Markov random process, while the two others are ordinary differential equations (ODEs) describing the evolution of mRNA and protein levels. Interactions between genes and stimulus are then characterized by the assumption that k_{on} and k_{off} are functions of $P = (P_1, \ldots, P_G)$ and Q. The form for k_{on} is the following (for k_{off}, replace $\theta_{i,j}$ by $-\theta_{i,j}$):

$$k_{\text{on}}(P,Q) = \frac{k_{\text{on\_min},i} + k_{\text{on\_max},i}\beta_i\Phi_i(P,Q)}{1 + \beta_i\Phi_i(P,Q)} \tag{2}$$

$$\Phi_i(P,Q) = \frac{1 + e^{\theta_{i,0}}Q}{1+Q} \prod_{j=1}^{G} \frac{1 + e^{\theta_{i,j}}\left(\frac{P_j}{H_j}\right)^{\gamma}}{1 + \left(\frac{P_j}{H_j}\right)^{\gamma}} \tag{3}$$

This interaction function slightly differs from [36] since auto-feedback is considered as any other interactions and stimulus effect is explicitly defined. Exponent parameter γ is set to default value 2. Interaction threshold H_j is associated to protein j. Interaction parameters $\theta_{i,j}$ will be estimated during the iterative inference. Parameter β_i corresponds to GRN external and constant influence on gene to define its basal expression: it is computed at simulation initialization in order to set k_{on} and k_{off} to their initial value. From now on, we drop the index i to simplify our notation when there is no ambiguity.

Overview of WASABI workflow

WASABI framework is divided in 3 main steps. First, individual gene parameters defined in model (1) (all except θ and H) are estimated before network inference from a number of experimental data types acquired during T2EC differentiation. They include time stamped single-cell transcriptomic [37], bulk transcription inhibition kinetic [37] and bulk proteomic data [39]. In a second step, genes are sorted regarding their wave times (see "Results" section for a description of wave concept) estimated from the mean of single cell transcriptomic data for promoter waves, and bulk proteomic data for protein waves. Finally, network iterative inference step is performed from single transcriptomic data, previously inferred gene parameters and sorted genes list. All methods are detailed in following sections, an overview of workflow is given by Fig 8.

For T2EC *in vitro* application, tables of gene parameters and wave times are provided in supporting information. For *in silico* benchmarking we assume that gene parameters d_0, d_1, s_1 are known. Single-cell data and bulk proteomic data are simulated from *in silico* GRNs for time points 0, 2, 4 ,8, 24, 33, 48, 72 and 100h.

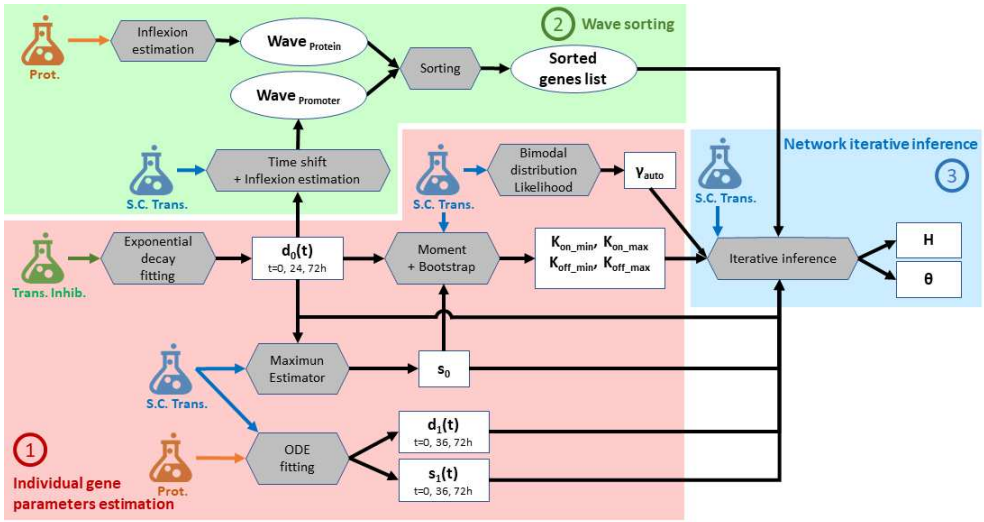


Fig 8. Parameters estimation workflow. Schematic view of WASABI workflow with 3 main steps: (1) individual gene parameters estimation (red zone), (2) waves sorting (green zone) and (3) network iterative interaction inference (blue zone). Wave concept is introduced in "Result" section. Model parameters (square boxes) are estimated from experimental data (flasks) with a specific method (grey hexagones). All methods are detailed in "Method" section. Estimated data relative to waves are represented by round boxes. Input arrows represent data required by methods to compute parameters. There are 3 types of experimental data, (i) bulk transcription inhibition kinetic (green flask), (ii) single-cell transcriptomic (blue flask) and (iii) proteomic data (orange flask). Model parameters are specific to each gene, except for θ, which is specific to a pair of regulator/regulated genes. Notations are consistent with Eq(1), γ_{auto} represents exponent term of auto-positive feedback interaction. Only $d_0(t)$, $d_1(t)$ and $s_1(t)$ are time dependent. One gene can have several wave times.

First step - Individual gene parameters estimation

Exponential decay fitting for mRNA degradation rate (d_0) estimation

The degradation rate d_0 corresponds to active decay (i.e. destruction of mRNA) plus dilution due to cell division. The RNA decay was already estimated in [37] before differentiation (0h), 24h and 72h after differentiation induction from population-based data of mRNA decay kinetic using actinomycin D-treated T2EC (`osf.io/k2q5b`). Cell division dilution rate is assumed to be constant during the differentiation process and cell cycle time has been experimentally measured at 20h [38].

Maximum estimator for mRNA transcription rate (s_0) estimation

To infer the transcription rate s_0, we used a maximum estimator based on single-cell expression data generated in [37]. We suppose that the highest possible mRNA level is given by s_0/d_0. Thus s_0 corresponds to the maximum mRNA count observed in all cells and time points multiplied by $\max_t(d_0(t))$.

Method of moments and bootstrapping for range of promoter switching rates ($k_{\text{on/off\_min/max}}$) estimation

Dynamic parameters k_{on} and k_{off} are bounded respectively by constant parameters $[k_{\text{on\_min}}; k_{\text{on\_max}}]$ and $[k_{\text{off\_min}}; k_{\text{off\_max}}]$ (see Eq (2)) which are estimated as follows from time course single-cell transcriptomic data. Parameters s_0 and $d_0(t)$ are supposed to be previously estimated for each gene at time t.

Range parameters shall be compliant with constraints (Eq (4)) imposed by the transcription dynamic regime observed *in vitro*. RNA distributions [37] have many zeros, which is consistent with the bursty regime of transcription. There is no observed RNA saturation in distributions. Moreover, all GRN parameters should also comply with computational constraints. On the one hand, the time step dt used for simulations shall be small enough regarding GRN dynamics to avoid aliasing (under-sampling) effects. On the other hand, dt should not be too small to save computation time. These constraints correspond to

$$k_{\text{on}} < d_0 < k_{\text{off}} < \frac{1}{dt} \tag{4}$$

and we deduce inequalities for ranges:

$$k_{\text{on\_min}} < k_{\text{on\_max}} < d_0 < k_{\text{off\_min}} < k_{\text{off\_max}} < \frac{1}{dt}. \tag{5}$$

We set the default value $k_{\text{on\_min}}$ to 0.001 h^{-1}. Parameter $k_{\text{on\_max}}$ is estimated from time course single-cell transcriptomic data after removing zeros. This truncation mimics a distribution where gene is always activated, so that k_{on} is close to its maximum value $k_{\text{on\_max}}$. With these truncated distributions, for each time point t, we estimate $k_{\text{on},t}$ using a moment-based method defined in [54]. We bootstrapped 1000 times to get a list of $k_{\text{on},t,n}$ with index n corresponding to bootstrap sample n. For each time point we compute the 95% percentile of $k_{\text{on},t,n}$, then we consider the mean value of these percentiles to have a first estimate of $k_{\text{on\_max}}$. This $k_{\text{on\_max}}$ is then down and up limited respectively between $k_{\text{on\_max\_lim\_min}}$ and $k_{\text{on\_max\_lim\_max}}$ given in Eq (6) to guarantee that observed k_{on} can be easily reached during simulations with reasonable values of protein level (because of asymptotic behavior of interaction function). In other words $k_{\text{on\_max}}$ shall not be too close from minimum or maximum observed k_{on} considering 10% margins. Finally, this limited $k_{\text{on\_max}}$ is up-limited by $0.5 \times \max_t(d_0(t))$ to guarantee a 50% margin with $d_0(t)$.

$$\begin{aligned} k_{\text{on\_max\_lim\_min}} &= \frac{\max_t(\text{median}_n(k_{\text{on},t,n})) - 0.1 \times k_{\text{on\_min}}}{0.9} \\ k_{\text{on\_max\_lim\_max}} &= \frac{\max_t(\text{median}_n(k_{\text{on},t,n})) - 0.9 \times k_{\text{on\_min}}}{0.1} \end{aligned} \tag{6}$$

Parameter $k_{\text{off\_min}}$ is set to $\max_t(d_0(t))$ to comply with equation Eq (5). Parameter $k_{\text{off\_max}}$ is estimated like $k_{\text{on\_max}}$ from time course single-cell transcriptomic data but without zero truncation.For each time point t, we estimate $k_{\text{off},t}$ using a moment-based method defined in [54]. We bootstrapped 1000 times to get a list of $k_{\text{off},t,n}$ with index n corresponding to bootstrap sample n. For each time point we compute the 95% percentile of $k_{\text{off},t,n}$, then we consider the mean value of these percentiles to have a first estimate of $k_{\text{off\_max}}$. This $k_{\text{off\_max}}$ is then down and up limited respectively between $k_{\text{off\_max\_lim\_min}}$ and $k_{\text{off\_max\_lim\_max}}$ given in Eq (7) to guarantee that observed k_{off} can be easily reached during simulations with reasonable values of protein level (because of asymptotic behavior of interaction function). In other words $k_{\text{off\_max}}$ shall not be too close from minimum or maximum observed k_{off} considering 10% margins. Finally, this limited $k_{\text{off\_max}}$ is up-limited by $1/dt$ to guaranty simulation anti-aliasing.

$$\begin{aligned} k_{\text{off\_max\_lim\_min}} &= \frac{\max_t(\text{median}_n(k_{\text{off},t,n})) - 0.1 \times k_{\text{off\_min}}}{0.9} \\ k_{\text{off\_max\_lim\_max}} &= \frac{\max_t(\text{median}_n(k_{\text{off},t,n})) - 0.9 \times k_{\text{off\_min}}}{0.1} \end{aligned} \tag{7}$$

ODE fitting for protein translation and degradation rates (d_1, s_1) estimation

Rates $d_1(t)$ and $s_1(t)$ are estimated from comparison of proteomic population kinetic data [39] with RNA mean value kinetic data computed from single-cell data [37]. Parameter $d_1(t)$ corresponds to protein active decay rate while total protein degradation rate $d_{1\_tot}(t)$ includes decay plus cell division dilution. Associated total protein half-life is referred to as $t_{1\_tot}(t)$. Parameters $s_1(t)$ and $d_{1\_tot}(t)$ are estimated using a calibration algorithm based on a maximum likelihood estimator (MLE) from package [55]. Objective function is given by the Root Mean Squared Error function (provided by the package) comparing experimental protein counts with simulated ones given by ODEs from our model (1) with RNA level provided by experimental mean RNA data:

$$P'(t) = s_1(t)M(t) - d_1(t)P(t)$$

52 out of our 90 selected genes were detected in proteomic data. 23 of these fit correctly experimental data with a constant d_1 and s_1 during differentiation. 5 genes were estimated with a variable $s_1(t)$ and a constant d_1 to fit a constant protein level with a decreasing RNA level. For the remaining 24 genes, protein level decreased while RNA is constant, which is modeled with s_1 constant and $d_1(t)$ variable.

For the genes that were not detected in our proteomic data we turned to the literature [56] and found 13 homologous genes with associated estimation of d_1 and s_1. For the remaining 25 genes, we estimated parameters with the following rationale: we consider that the non-detection in the proteomic data is due to low protein copy number, lower than 100. Moreover [56] proposed an exponential relation between s_1 and the mean protein level that we confirmed with our data (see supporting information), resulting in the following definition:

$$s_1 = 10^{-1.47} \times P^{0.81}$$

Linear regression was performed using the Python scipy.stats.linregress() method from Scipy package with the following parameters: $r^2 = 0.55$, slope $= 0.81$, intercept $= -1.47$ and $p = 2.97 \times 10^{-9}$. Therefore, if we extrapolate this relation for low protein copy numbers assuming $P < 100$ copies, s_1 should be lower than 1 molecule/RNA/hour. Assuming the relation

$$\text{Prot} = \text{RNA} \times \frac{s_1}{d_{1\_tot}}$$

between mean protein and RNA levels, we deduced a minimum value of d_1 from mean RNA level given by: $d_1 > \text{RNA}/100$. We set s_1 and d_1 respectively to their maximum and minimum estimated values.

Bimodal distribution likelihood for auto-positive feedback exponent (γ_{auto}) estimation

We inferred the presence of auto-positive feedback by fitting an individual model for each gene, based on [36]. The model is characterized by a Hill-type power coefficient. The value of this coefficient was inferred by maximizing the model likelihood, available in explicit form. The key idea is that genes with auto-positive feedback typically show, once viewed on an appropriate scale, a strongly bimodal distribution during their transitory regime. The interested reader may find some details in the supplementary information file of [36], especially in sections 3.6 and 5.2. Note that such auto-positive feedback may reflect either a direct auto-activation, or a strong but indirect positive loop, potentially involving other genes. Estimated Hill-type power coefficients for *in silico* and *in vitro* networks are provided in supporting information.

Second step - Waves sorting

Inflexion estimator for wave time estimation

Wave time for gene promoter W_{prom} and protein W_{prot} are estimated regarding their respective mean trace $\overline{E}$ and $\overline{P}$. Estimation differs depending on mean trace monotony. *In vitro* wave times are provided in supporting information.

1) If the mean trace is monotonous (checked manually), it is smoothed by a 3rd order polynomial approximation using method poly1d() from python numpy package. Wave time is then defined as the inflection time point of polynomial function where 50% of evolution between minimum and maximum is reached.

2) If the mean trace is not monotonous, it is approximated by a piecewise-linear function with 3 breakpoints that minimizes the least square error. Linear interpolations are performed using the polynomial.polyfit() function from python numpy package. Selection of breakpoints is performed using optimize.brute() function from python numpy package.

We obtained a series of 4 segments with associated breakpoints coordinate and slope. Slopes are thresholded: if absolute value is lower than 0.2 it is considered null. Then, we looked for inflection break times where segments with non null slope have an opposite sign compare to the previous segment, or if previous segment has a null slope. Each inflection break time corresponds to an initial effect of a wave. A valid time, when wave effect applies, is associated and corresponds to next inflection break time or to the end of differentiation. Thus, we obtained couples of inflection break time and valid time which defined the temporal window of associated wave effect. For each wave window, if mean trace variation between inflection break time and valid time is large enough (i.e., greater than 20% of maximal variation during all differentiation process for the gene), a wave time is defined as the time where half of mean trace variation is reached during wave time window.

Protein mean trace $\overline{P}$ is given by proteomic data if available, else it is computed from simulation traces with 500 cells using the model with the parameters estimated earlier. Promoter mean trace $\overline{E}$ is computed as follows from mean RNA trace (from single-cell transcriptomic data) with time delay correction induced by mRNA degradation rate d_0.

$$\overline{E}(t) = \frac{k_{\text{on}}(t)}{k_{\text{on}}(t) + k_{\text{off}}(t)}$$

$$\overline{E}\left(t - \frac{1}{d_0(t)}\right) = \frac{d_0}{s_0} \times \overline{M}(t) \times \left(t - \frac{1}{d_0(t)}\right)$$

Genes sorting

Genes are sorted regarding their promoter waves time W_{prom}. Genes with multiple waves, in case of feedback for example, are present several times in the list. Moreover, genes are classified by groups regarding their position in the network. Genes directly regulated by the stimulus are called the early genes; Genes that regulates other genes are defined as regulatory genes; Genes that do not influence other genes are identified as readout genes. Note that genes can belong to several group.

We can deduce the group type for each gene from its wave time estimation. Subsequent constraints have been defined from *in silico* benchmarking (see Results section). A gene i belongs to one of these groups according to following rules:

- if $W_{prom} < 5$h then it is an early gene
- if $W_{prom} < 7$h then it could be an early gene or another types
- if $\max_i(W_{prom,i}) + 30\text{h} < W_{prot}$ then it is a readout gene
- else it could be a regulatory or a readout gene

Third step - Network iterative inference

Interaction threshold (H)

Interaction threshold H is estimated for each protein. It corresponds to mean protein level at 25% between minimum and maximum mean protein level observed during differentiation by *in silico* simulations:

$$H = P_{\min} + 0.25(P_{\max} - P_{\min})$$

We choose the value of 25% to maximize the amplitude variation of k_{on} and k_{off} of gene target induced by the shift of the regulator protein level from its minimal to maximal value (see Eq(2)).

Iterative calibration algorithm ($\theta_{i,j}$)

The following algorithm gives a global overview of the iterative inference process:

Generate_EARLY_network(): In a first step we calibrate the interactions between early genes and stimulus ($\theta_{i,0}$) to obtain an initial sub-GRN. Calibration algorithm *Calibrate()* is defined below.

List_genes_sorted_by_Wave_time: This list is computed prior to iterative inference (see previous subsection).

Algorithm 1 WASABI GRN iterative inference

```
List_GRN_candidates = Generate_EARLY_network()
for Gene, Wave in List_genes_sorted_by_Wave_time do
    for GRN in List_GRN_candidates() do
        List_new_GRN_to_calibrate = Get_all_possible_interaction(GRN, Gene, Wave)
        for New_GRN in New_GRN_List do
            Calibrate(New_GRN)
    List_GRN_candidate = Select_Best_New_GRN()
```

Get_all_possible_interaction(GRN, Gene, Wave): For each GRN candidate we estimate all possible interactions with the new gene and prior regulatory genes, or stimulus, regarding their respective promoter wave and protein wave with the following logic: if promoter wave is lower than 7h, interaction is possible between stimulus and the new gene. If the difference of promoter wave minus protein wave is between −20h and +30h, then there is a possible interaction between the new gene and regulatory gene. Note: if WASABI is run in "directed" mode, only the true interaction is returned.

Calibrate(New_GRN): For interaction parameter calibration we used a Maximum Likelihood Estimator (MLE) from package spotpy [55]. The goal is to fit simulated single-cell gene marginal distribution with *in vitro* ones tuning efficiency interaction parameter $\theta_{i,j}$. For *in silico* study we defined **GRN Fit distance** as the mean of the 3 worst gene-wise fit distances. For *in vitro* study we defined **GRN Fit distance** as the mean of the fit distances of all genes. Gene-wise fit distance is defined as the mean of the 3 higher Kantorovitch distances [42] among time points. For a given time point and a given gene, the Kantorovitch fit distance corresponds to a distance between marginal distributions of simulated and experimental expression data. At the end of calibration the set of interaction parameter $\theta_{i,j}$ with associated GRN Fit distance is returned.

Select_Best_New_GRN() We fetch all GRN calibration fitting outputs from remote servers and select best new GRNs to be expanded for next iteration updating list of List_GRN_candidate. New networks candidates are limited by number of available computational cores.

GRN simulation

We use a basic Euler solver with fixed time step ($dt = 0.5$h) to solve mRNA and protein ODEs [36]. The promoter state evolution between t and $t + dt$ is given by a Bernoulli distributed random variable

$$E(t + dt) = \text{Bernoulli}(p(t))$$

drawn with probability $p(t)$ depending on current k_{on}, k_{off} and promoter state:

$$p(t) = E(t)e^{-dt(k_{\text{on}}+k_{\text{off}})} + \frac{k_{\text{on}}}{k_{\text{on}} + k_{\text{off}}}\left(1 - e^{-dt(k_{\text{on}}+k_{\text{off}})}\right).$$

Time-dependent parameters like d_0, d_1 and s_1 are linearly interpolated between 2 points. The stimulus Q is represented by a step function between 0 and 1000 at $t = 0$h. Simulation starts at $t = -60$h to ensure convergence to steady state before the stimulus is applied. Parameters k_{on} and k_{off} are given by Eq (2).

Acknowledgments

We thank the computational center of IN2P3 (Villeurbanne/France), specially Pascal Calvat, for access to HPC facilities; Eddy Caron (Avalon, ENS Lyon/INRIA) for his support on parallel computing implementation; Patrick Mayeux for proteomic data; and Rudiyanto Gunawan (ETH, Zürich) for critical reading of the manuscript. We would like to thank all members of the SBDM team, Dracula team, and Camilo La Rota (Cosmotech) for enlightening discussions, We also thank the BioSyL Federation and the Ecofect Labex (ANR-11-LABX-0048) of the University of Lyon for inspiring scientific events.

Funding

This work was supported by funding from the French agency ANR (ICEBERG; ANR-IABI-3096 and SinCity; ANR-17-CE12-0031) and the Association Nationale de la Recherche Technique (ANRT, CIFRE 2015/0436). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

References

1. MacNeil LT, Walhout AJ. Gene regulatory networks and the role of robustness and stochasticity in the control of gene expression. Genome research. 2011;21(5):645–657.
2. Greene JA, Loscalzo J. Putting the Patient Back Together - Social Medicine, Network Medicine, and the Limits of Reductionism. N Engl J Med. 2017;377(25):2493–2499.
3. Sugimura R, Jha DK, Han A, Soria-Valles C, da Rocha EL, Lu YF, et al. Haematopoietic stem and progenitor cells from human pluripotent stem cells. Nature. 2017;545:432.
4. Lis R, Karrasch CC, Poulos MG, Kunar B, Redmond D, Duran JGB, et al. Conversion of adult endothelium to immunocompetent haematopoietic stem cells. Nature. 2017;545:439.
5. Ieda M, Fu JD, Delgado-Olguin P, Vedantham V, Hayashi Y, Bruneau BG, et al. Direct reprogramming of fibroblasts into functional cardiomyocytes by defined factors. Cell. 2010;142(3):375–386.
6. Madhamshettiwar PB, Maetschke SR, Davis MJ, Reverter A, Ragan MA. Gene regulatory network inference: evaluation and application to ovarian cancer allows the prioritization of drug targets. Genome Med. 2012;4(5):41.
7. Creixell P, Schoof EM, Erler JT, Linding R. Navigating cancer network attractors for tumor-specific therapy. Nature biotechnology. 2012;30(9):842.
8. Chai LE, Loh SK, Low ST, Mohamad MS, Deris S, Zakaria Z. A review on the computational approaches for gene regulatory network construction. Comput Biol Med. 2014;48:55–65.
9. Dojer N, Gambin A, Mizera A, Wilczyński B, Tiuryn J. Applying dynamic Bayesian networks to perturbed gene expression data. BMC Bioinformatics. 2006;7(1):249.

10. Vinh NX, Chetty M, Coppel R, Wangikar PP. Gene regulatory network modeling via global optimization of high order Dynamic Bayesian Networks. BMC Bioinf. 2012;27:2765–2766.

11. Akutsu T, Miyano S, Kuhara S. Identification of genetic networks from a small number of gene expression pattern under the Boolean model. Pacific Symp Biocomput. 1999;4:17–28.

12. Saadatpour A, Albert R. Boolean modeling of biological regulatory networks: A methodology tutorial. Methods. 2013;62(1):3–12.

13. Zhao W, Serpedin E, Dougherty ER. Inferring gene regulatory networks from time series data using the minimum description length principle. Bioinformatics. 2006;22(17):2129–2135.

14. Polynikis A, Hogan SJ, di Bernardo M. Comparing different ODE modelling approaches for gene regulatory networks. J Theor Biol. 2009;261(4):511–30.

15. Bansal M, Belcastro V, Ambesi-Impiombato A, Di Bernardo D. How to infer gene networks from protein profiles. Mol Syst Biol. 2007;3:1–10.

16. Svensson V, Natarajan KN, Ly LH, Miragaia RJ, Labalette C, Macaulay IC, et al. Power analysis of single-cell RNA-sequencing experiments. Nature Methods. 2017;14:381.

17. Fiers M, Minnoye L, Aibar S, Bravo Gonzalez-Blas C, Kalender Atak Z, Aerts S. Mapping gene regulatory networks from single-cell omics data. Brief Funct Genomics. 2018;.

18. Babtie A, Chan TE, Stumpf MPH. Learning regulatory models for cell development from single cell transcriptomic data. Current Opinion in Systems Biology. 2017;5:72D81.

19. Yvert G. 'Particle genetics': treating every cell as unique. Trends Genet. 2014;30(2):49–56.

20. Dueck H, Eberwine J, Kim J. Variation is function: Are single cell differences functionally important?: Testing the hypothesis that single cell variation is required for aggregate function. Bioessays. 2016;38(2):172–80.

21. Symmons O, Raj A. What's Luck Got to Do with It: Single Cells, Multiple Fates, and Biological Nondeterminism. Mol Cell. 2016;62(5):788–802.

22. Cannoodt R, Saelens W, Saeys Y. Computational methods for trajectory inference from single-cell transcriptomics. Eur J Immunol. 2016;46(11):2496–2506.

23. Chen H, Guo J, Mishra SK, Robson P, Niranjan M, Zheng J. Single-cell transcriptional analysis to uncover regulatory circuits driving cell fate decisions in early mouse development. Bioinformatics. 2015;31(7):1060–6.

24. Lim CY, Wang H, Woodhouse S, Piterman N, Wernisch L, Fisher J, et al. BTR: training asynchronous Boolean models using single-cell expression data. BMC Bioinformatics. 2016;17(1):355.

25. Moignard V, Woodhouse S, Haghverdi L, Lilly AJ, Tanaka Y, Wilkinson AC, et al. Decoding the regulatory network of early blood development from single-cell gene expression measurements. Nat Biotechnol. 2015;33(3):269–76.

26. Matsumoto H, Kiryu H. SCOUP: a probabilistic model based on the Ornstein-Uhlenbeck process to analyze single-cell expression data during differentiation. BMC Bioinformatics. 2016;17(1):232.

27. Cordero P, Stuart JM. Tracing co-regulatory network dynamics in noisy, single-cell transcriptome trajectories. Biocomputing. 2017; p. 576–587.

28. Sanchez-Castillo M, Blanco D, Tienda-Luna IM, Carrion MC, Huang Y. A Bayesian framework for the inference of gene regulatory networks from time and pseudo-time series data. Bioinformatics. 2017;.

29. Matsumoto H, Kiryu H, Furusawa C, Ko MSH, Ko SBH, Gouda N, et al. SCODE: an efficient regulatory network inference algorithm from single-cell RNA-Seq during differentiation. Bioinformatics. 2017;33(15):2314–2321.

30. Ocone A, Haghverdi L, Mueller NS, Theis FJ. Reconstructing gene regulatory dynamics from high-dimensional single-cell snapshot data. Bioinformatics. 2015;31(12):i89–i96.

31. Huang S. Non-genetic heterogeneity of cells in development: more than just noise. Development. 2009;136(23):3853–62.

32. Sokolik C, Liu Y, Bauer D, McPherson J, Broeker M, Heimberg G, et al. Transcription factor competition allows embryonic stem cells to distinguish authentic signals from noise. Cell Syst. 2015;1(2):117–129.

33. Munsky B, Trinh B, Khammash M. Listening to the noise: random fluctuations reveal gene network parameters. Mol Syst Biol. 2009;5:318.

34. Moris N, Pina C, Arias AM. Transition states and cell fate decisions in epigenetic landscapes. Nat Rev Genet. 2016;17(11):693–703.

35. Papili Gao N, Ud-Dean SMM, Gandrillon O, Gunawan R. SINCERITIES: Inferring gene regulatory networks from time-stamped single cell transcriptional expression profiles. Bioinformatics. 2018;34(2):258–266.

36. Herbach U, Bonnaffoux A, Espinasse T, Gandrillon O. Inferring gene regulatory networks from single-cell data: a mechanistic approach. BMC Systems Biology. 2017;11(1).

37. Richard A, Boullu L, Herbach U, Bonnaffoux A, Morin V, Vallin E, et al. Single-Cell-Based Analysis Highlights a Surge in Cell-to-Cell Molecular Variability Preceding Irreversible Commitment in a Differentiation Process. PLoS Biol. 2016;14(12):e1002585.

38. Gandrillon O, Schmidt U, Beug H, Samarut J. TGF-beta cooperates with TGF-alpha to induce the self-renewal of normal erythrocytic progenitors: evidence for an autocrine mechanism. Embo J. 1999;18(10):2764–2781.

39. Leduc M, Gautier EF, Guillemin A, Broussard C, Salnot V, Lacombe C, et al. Deep proteomic analysis of chicken erythropoiesis. bioRxiv preprint. 2018;.

40. Liu Z, Tjian R. Visualizing transcription factor dynamics in living cells. J Cell Biol. 2018;.

41. Lambert SA, Jolma A, Campitelli LF, Das PK, Yin Y, Albu M, et al. The Human Transcription Factors. Cell. 2018;172:650–665.

42. Baba A, Komatsuzaki T. Construction of effective free energy landscape from single-molecule time series. Proc Natl Acad Sci U S A. 2007;104(49):19297–302.

43. Croubois H, Bonnaffoux A, Gandrillon O, Caron E. A Cloud-aware Autonomous Workflow Engine and Its Application to Gene Regulatory Networks Inference. Conference: Closer 2018. 2018; p. 8.

44. Olsen JV, Mann M. Status of large-scale analysis of post-translational modifications by mass spectrometry. Mol Cell Proteomics. 2013;12(12):3444–52.

45. Manning KS, Cooper TA. The roles of RNA processing in translating genotype to phenotype. Nat Rev Mol Cell Biol. 2017;18(2):102–114.

46. Mandic A, Strebinger D, Regali C, Phillips NE, Suter DM. A novel method for quantitative measurements of gene expression in single living cells. Methods. 2017;120:65–75.

47. Lin YT, Hufton PG, Lee EJ, Potoyan DA. A stochastic and dynamical view of pluripotency in mouse embryonic stem cells. PLoS Comput Biol. 2018;14(2):e1006000.

48. Ud-Dean SM, Gunawan R. Optimal design of gene knockout experiments for gene regulatory network inference. Bioinformatics. 2016;32(6):875–83.

49. Kreutz C, Timmer J. Systems biology: experimental design. FEBS J. 2009;276(4):923–42.

50. Semrau S, Goldmann JE, Soumillon M, Mikkelsen TS, Jaenisch R, van Oudenaarden A. Dynamics of lineage commitment revealed by single-cell transcriptomics of differentiating embryonic stem cells. Nature Communications. 2017;8(1).

51. Jang S, Choubey S, Furchtgott L, Zou LN, Doyle A, Menon V, et al. Dynamics of embryonic stem cell differentiation inferred from single-cell transcriptomics show a series of transitions through discrete cell states. Elife. 2017;6.

52. Barabasi AL, Oltvai ZN. Network biology: understanding the cell's functional organization. Nat Rev Genet. 2004;5(2):101–13.

53. Hart Y, Alon U. The utility of paradoxical components in biological circuits. Mol Cell. 2013;49(2):213–21.

54. Peccoud J, Ycart B. Markovian Modelling of Gene Product Synthesis. Theoretical population biology. 1995;48:222–234.

55. Houska T, Kraft P, Chamorro-Chavez A, Breuer L. SPOTting Model Parameters Using a Ready-Made Python Package. PLOS ONE. 2015;10(12):e0145180.

56. Schwanhausser B, Busse D, Li N, Dittmar G, Schuchhardt J, Wolf J, et al. Corrigendum: Global quantification of mammalian gene expression control. Nature. 2013;495(7439):126–127.

Chapitre 6

Application de l'approximation de Hartree

Comme les données dont on dispose contiennent bien des dépendances statistiques non triviales entre les gènes, il est assez naturel de chercher à utiliser ces dépendances pour l'inférence : cela motive des méthodes exploitant la loi jointe et en particulier l'approximation de Hartree présentée dans la section 2.2, contrairement à l'approche du chapitre précédent qui n'utilise que les lois marginales de chaque gène.

Dans ce chapitre, nous abordons la mise en pratique de cette approximation de Hartree. Malgré le fait qu'il ne s'agisse que d'une pseudo-vraisemblance, il est possible de la maximiser exactement comme une vraisemblance classique, à ceci près que l'on utilise un algorithme de type EM puisque seul l'ARNm est observé, les protéines devant être considérées comme des variables latentes. La méthode que nous avons retenue consiste en fait, en notant x le vecteur des ARNm et y celui des protéines,[1] à maximiser la pseudo-vraisemblance complète $p_\theta(x, y)$ alternativement en θ et en y : cet algorithme est parfois appelé *classification EM* (Celeux et Govaert, 1991). Par ailleurs, au lieu d'utiliser l'estimateur du maximum de vraisemblance classique, on considère l'estimateur du *maximum a posteriori* (MAP) après avoir introduit une certaine distribution a priori sur la matrice θ, ce qui correspond à pénaliser $\ln p_\theta(x, y)$. Cette pénalisation prend une forme assez spéciale et permet d'avoir des zéros exacts dans la matrice ainsi θ renvoyée, de façon à inférer clairement la présence ou l'absence d'une interaction dans le réseau, tout en mettant en compétition les paramètres θ_{ij} et θ_{ji} pour ne garder les deux que si les données le suggèrent vraiment. Ces aspects sont décrits dans l'appendice 6.A, qui détaille également le principe de l'algorithme proximal utilisé pour effectuer les maximisations successives en θ.

Enfin, étant associée au modèle de chromatine du chapitre 1, la méthode nécessite la connaissance des hyperparamètres $k_{0,i}$, $k_{1,i}$, s_{ij} et m_{ij}. La Remarque 2.23 suggère une phase préliminaire basée sur l'auto-modèle Gamma-Binomial dans le cas de gènes indépendants (cf. appendice 6.A.5). Noter que l'on peut seulement inférer $k_{0,i}/d_{0,i}$, $k_{1,i}/d_{0,i}$ et m_{ij} : la connaissance des rapports $d_{0,i}/d_{1,i}$ constitue le minimum requis pour pouvoir appliquer la méthode, et les paramètres s_{ij} ne sont pas identifiables mais peuvent être fixés à des valeurs cohérentes. Dans l'article qui suit, on effectue un test de performance sur des données simulées par le modèle PDMP : la phase préliminaire n'est donc pas nécessaire et l'on préfère se baser sur les bonnes valeurs des hyperparamètres pour inférer θ.

1. On prendra garde à l'interversion des notations par rapport aux deux premiers chapitres : dans l'article qui suit, x désigne l'ARNm et y désigne les protéines.

Herbach *et al. BMC Systems Biology* (2017) 11:105
DOI 10.1186/s12918-017-0487-0

BMC Systems Biology

RESEARCH ARTICLE Open Access

Inferring gene regulatory networks from single-cell data: a mechanistic approach

Ulysse Herbach[1,2,3], Arnaud Bonnaffoux[1,2,4], Thibault Espinasse[3] and Olivier Gandrillon[1,2*]

Abstract

Background: The recent development of single-cell transcriptomics has enabled gene expression to be measured in individual cells instead of being population-averaged. Despite this considerable precision improvement, inferring regulatory networks remains challenging because stochasticity now proves to play a fundamental role in gene expression. In particular, mRNA synthesis is now acknowledged to occur in a highly bursty manner.

Results: We propose to view the inference problem as a fitting procedure for a mechanistic gene network model that is inherently stochastic and takes not only protein, but also mRNA levels into account. We first explain how to build and simulate this network model based upon the coupling of genes that are described as piecewise-deterministic Markov processes. Our model is modular and can be used to implement various biochemical hypotheses including causal interactions between genes. However, a naive fitting procedure would be intractable. By performing a relevant approximation of the stationary distribution, we derive a tractable procedure that corresponds to a statistical hidden Markov model with interpretable parameters. This approximation turns out to be extremely close to the theoretical distribution in the case of a simple toggle-switch, and we show that it can indeed fit real single-cell data. As a first step toward inference, our approach was applied to a number of simple two-gene networks simulated in silico from the mechanistic model and satisfactorily recovered the original networks.

Conclusions: Our results demonstrate that functional interactions between genes can be inferred from the distribution of a mechanistic, dynamical stochastic model that is able to describe gene expression in individual cells. This approach seems promising in relation to the current explosion of single-cell expression data.

Keywords: Single-cell transcriptomics, Gene network inference, Multiscale modelling, Piecewise-deterministic Markov processes

Background

Inferring regulatory networks from gene expression data is a longstanding question in systems biology [1], with an active community developing many possible solutions. So far, almost all studies have been based on population-averaged data, which historically used to be the only possible way to observe gene expression. Technologies now allow us to measure mRNA levels in individual cells [2–4], a revolution in terms of precision. However, the network reconstruction task paradoxically remains more challenging than ever.

The main reason is that the variability in gene expression unexpectedly stands at a large distance from a trivial, limited perturbation around the population mean. It is now clear indeed that this variability can have functional significance [5–7] and should therefore not be ignored when dealing with gene network inference. In particular, as the mean is not sufficient to account for a population of cells, a deterministic model – e.g. ordinary differential equation (ODE) systems, often used in inference [8, 9] – is unlikely to faithfully inform about an underlying gene regulatory network. Whether such a deterministic approach could still be a valid approximation or not is a difficult question that may require some biological insight into the system under consideration [10]. Another key aspect when considering individual cells is that they generally have to be killed for measurements: from a statistical point of view, temporal single-cell data therefore should

*Correspondence: olivier.gandrillon@ens-lyon.fr
[1]Univ Lyon, ENS de Lyon, Univ Claude Bernard, CNRS UMR 5239, INSERM U1210, Laboratory of Biology and Modelling of the Cell, 46 allée d'Italie Site Jacques Monod, F-69007 Lyon, France
[2]Inria Team Dracula, Inria Center Grenoble Rhône-Alpes, Lyon, France
Full list of author information is available at the end of the article

not be seen as a set of time series, but rather *snapshots*, i.e. independent samples from a time series of distributions.

On the other hand, single-cell data give the opportunity of moving one step further toward a more accurate physical description of gene expression. Molecular processes of gene expression are overall now well understood, in particular transcription, but precisely how stochasticity emerges is still somewhat of a conundrum. Harnessing variability in single-cell data is expected to allow for the identification of critical parameters and also to provide hints about the basic molecular processes involved [11, 12]. Moreover, the variability arising from perturbations in cell populations is often crucial for network reconstruction to succeed [13, 14] as the deterministic inference problem suffers from intrinsic limitations [15]. From this point of view, the same information is expected to be contained in the variability between cells in single-cell data. Some of the few existing single-cell inference methods follow this path, for example using asynchronous Boolean network models [16] or generating pseudo time series [9, 17]. In this article, we use a mechanistic approach in the sense that every part of our model has an explicit physical interpretation. Importantly, mRNA observations are not used as a proxy for proteins since both are explicitly modeled.

Besides, mechanistic models that are accurate enough to describe gene expression at the single-cell level usually do not consider interactions between genes. For example, the so-called "two-state" (aka random telegraph) model has been successfully used with single-cell RNA-seq data [18], but the joint distribution of a set of genes contains much more information than the marginal kinetics of individual genes: our aim is to exploit this information while keeping the mechanistic point of view.

Namely, we propose to view the inference as a fitting procedure for a mechanistic gene network model. Whereas the goal here is not to achieve global predictability performances (e.g. as in [19]), our framework makes it possible to explicitly implement many biological hypotheses, and to test them by going back and forth between simulations and experiments. The main point of this article is to show that a tractable statistical model for network inference from single-cell data can be derived through successive relevant approximations. Finally, we demonstrate that our approach is capable of extracting enough information out of in silico-simulated noisy single-cell data to correctly infer the structures of various two-gene networks.

Methods

In this part, we aim at deriving a tractable statistical model from a mechanistic one. We will use the two-state model for gene expression to build a "network of two-state models" by making the promoter switching rates depend on protein levels. Then, successive relevant simplifications will lead to an explicit approximation of a statistical likelihood.

A simple mechanistic model for gene regulatory networks

Basic block: stochastic expression of a single gene

Our starting point is the well-known two-state model of gene expression [20–23], a refinement of the model introduced by [24] from pioneering single-cell experiments [25]. In this model, a gene is described by its promoter which can be either active (on) or inactive (off) – possibly representing a transcription complex being "bound" or "unbound" but it may be more complicated [26] – with mRNA being transcribed only during the active periods. Translation is added in a standard way, each mRNA molecule producing proteins at a constant rate. The resulting model (Fig. 1) can be entirely defined by the set of chemical reactions detailed in Table 1, where chemical species G, G^*, M and P respectively denote the inactive promoter, the active promoter, the amount of mRNA and proteins. The mathematical framework generally assumes stochastic mass-action kinetics [27] for all reactions, since they typically involve few molecules compared to Avogadro's number. In this fully discrete setting, one can use the master equation to compute stationary distributions: for mRNA the exact distribution is a Beta-Poisson mixture [28], and an approximation is available for proteins when they degrade much more slowly than mRNA [29]. In addition, the time-dependent generating function of mRNA is known in closed form [30] and can be inverted in some cases to obtain the transient distribution [28].

In practice, the formulas involve hypergeometric series that are not straightforward to use in a statistical inference framework. Besides, these series essentially arise from the fact that such a discrete model has to enumerate all potential collisions between molecules (the stochastic mass-action assumption in the master equation). It is therefore natural to consider keeping only the most important source of noise, that is, keeping a molecular representation for rare species but describing abundant

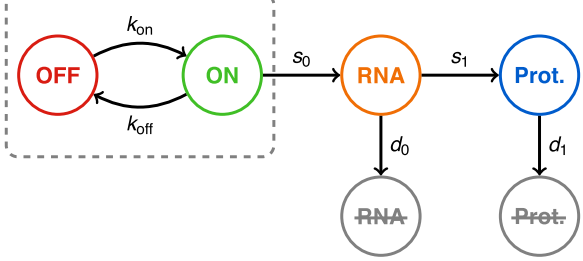


Fig. 1 Scheme of the two-state model of gene expression. We use it as the basic block of our network model

Table 1 Chemical reactions defining the two-state model. The rate constants are usually abbreviated to *rates* as they correspond to actual reactions rates when only one molecule of reactant is present. In the stochastic setting, these rates are in fact propensities, i.e. probabilities per unit of time

Reaction	Rate constant	Interpretation
$G \to G^*$	k_{on}	gene activation
$G^* \to G$	k_{off}	gene inactivation
$G^* \to G^* + M$	s_0	transcription
$M \to M + P$	s_1	translation
$M \to \varnothing$	d_0	mRNA degradation
$P \to \varnothing$	d_1	protein degradation

species at a higher level where molecular noise averages out to continuous quantities. A quick look at reactions in Table 1 indicates that the only rare species are G and G^*, with quantities $[G]$ and $[G^*]$ being equal to 0 or 1 molecule and satisfying the conservation relation $[G]+[G^*]= 1$. The other two, M and P, are not conserved quantities in the model and reach a much wider range in biological situations [31], meaning that saturation constants s_0/d_0 and s_1/d_1 are much larger than 1 molecule.

Hence, letting $E(t)$, $M(t)$ and $P(t)$ denote the respective quantities of G^*, M and P at time t, we consider a hybrid version of the previous model, where E has the same stochastic dynamics as before, but with M and P now following usual rate equations:

$$\begin{cases} E(t) : 0 \xrightarrow{k_{\text{on}}} 1, \quad 1 \xrightarrow{k_{\text{off}}} 0 \\ M'(t) = s_0 E(t) - d_0 M(t) \\ P'(t) = s_1 M(t) - d_1 P(t) \end{cases} \tag{1}$$

This system simply switches between two ordinary differential equations, depending on the value of the two-state continuous-time Markov process $E(t)$, making it a Piecewise-Deterministic Markov Process (PDMP) [32]. From a mathematical perspective, model (1) rigorously approximates the original molecular model when s_0/d_0 and s_1/d_1 are large enough [33, 34] and interestingly, it has already been implicitly considered in the biological literature [22, 23]. Note also that the stationary distribution of mRNA is a scaled Beta distribution that is exactly the one of the Beta-Poisson mixture in the discrete model [28]. Similarly to a recent approach for a two-gene toggle switch [35], we will use (1) as a basic building block for gene networks.

When both $k_{\text{on}} \ll k_{\text{off}}$ and $d_0 \ll k_{\text{off}}$, mRNA is transcribed by *bursts*, i.e. during short periods which make the mRNA quantity stay far from saturation. Hence, the amount transcribed within each burst is approximately proportional to the burst duration, whose mean is $1/k_{\text{off}}$ by definition: this justifies the quantity s/k_{off} often being called "burst size" or "burst amplitude". Furthermore, promoter active periods are much shorter than inactive ones so they can be seen as instantaneous, justifying the name "burst frequency" for the inverse of the mean inactive time k_{on}. We place ourselves in this situation as it often occurs in experiments [22, 23, 36–38]. Note however that these two notions are not clearly defined when relations $k_{\text{on}} \ll k_{\text{off}}$ and $d_0 \ll k_{\text{off}}$ do not hold.

Adding interactions between genes: the network model

Now considering a given set of n genes, a natural way of building a network is to assume that each gene i produces specific mRNA M_i and protein P_i, and to define a version of model (1) with its own parameters:

$$\begin{cases} E_i(t) : 0 \xrightarrow{k_{\text{on},i}} 1, \quad 1 \xrightarrow{k_{\text{off},i}} 0 \\ M_i'(t) = s_{0,i} E_i(t) - d_{0,i} M_i(t) \\ P_i'(t) = s_{1,i} M_i(t) - d_{1,i} P_i(t) \end{cases} \tag{2}$$

Still, genes have static parameters and do not interact with each other. To get an actual network, we need to go one step further: reactions $G_i \to G_i^*$ and $G_i^* \to G_i$ are not assumed to be elementary anymore, but rather represent complex reactions involving proteins so that promoter parameters $k_{\text{on},i}$ and $k_{\text{off},i}$ now depend on proteins (Fig. 2a), and a fortiori on time. Our network model will correspond to the explicit definition, for all gene i, of functions $k_{\text{on},i}(P_1, \ldots, P_n)$ and $k_{\text{off},i}(P_1, \ldots, P_n)$. These functions shall also depend on network-specific parameters quantifying the interactions, thus making the link between "fitting a chemical model" and "inferring a network". As a toy example, consider replacing $G_i \to G_i^*$ with two parallel elementary reactions

$$G_i \xrightarrow{\theta_{i,0}} G_i^* \quad \text{and} \quad G_i + P_j \xrightarrow{\theta_{i,j}} G_i^* + P_j \tag{3}$$

for which applying the law of mass action directly gives $k_{\text{on},i}(P_1, \ldots, P_n) = \theta_{i,0} + \theta_{i,j} P_j$. In a regulatory network (Fig. 2b), it would correspond to adding a directed edge from gene j to gene i, with $\theta_{i,0}$ the basal parameter of gene i, and $\theta_{i,j}$ the strength of activation of gene i by protein P_j. We emphasize that the action of P_j on the promoter G_i is not necessarily direct. For example, P_j can instead indirectly modulate the amount/activity of a transcription factor: we suppose in this article that such hidden reactions are fast enough regarding gene expression dynamics so that protein P_j is a relevant proxy for the transcription factor. Moreover, although we assume here that interactions can only happen at the level of $k_{\text{on},i}$ and $k_{\text{off},i}$, mainly for identifiability purposes, it is also possible to make $d_{1,i}$ and $s_{1,i}$ depend on proteins without fundamentally changing the mathematical approach (e.g. see [39, 40]).

In order to simplify notations, we normalize model (2) into a dimensionless equivalent model: we rewrite it in

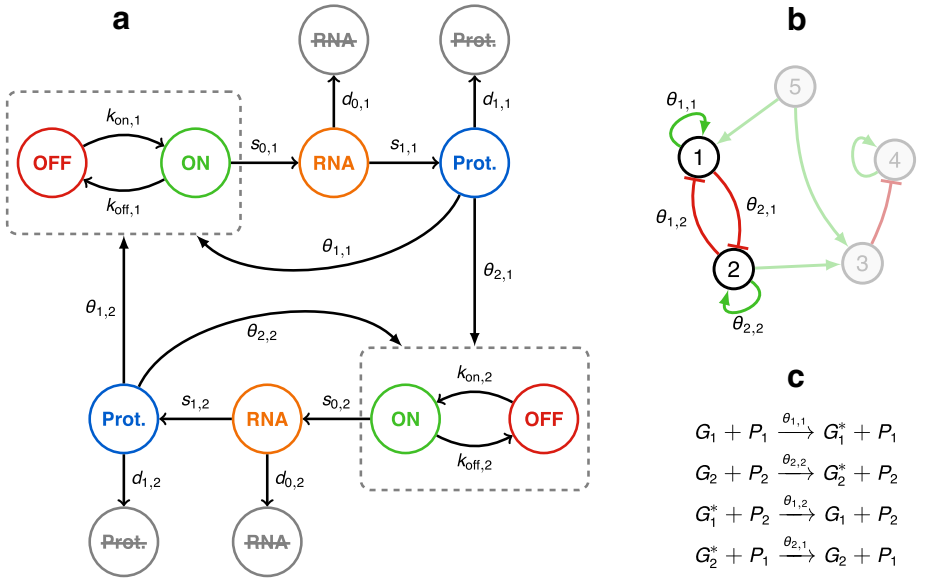


Fig. 2 Different views of the network model. **a** Two genes interacting with each other, forming a network. Interactions are assumed to arise from the dependence of promoter dynamics on protein quantities. **b** A higher level of abstraction leads to the traditional gene regulatory network representation. **c** A toy example of reactions defining the interactions between genes 1 and 2, making the link between representations (**a**) and (**b**)

terms of new variables $\overline{M}_i = \frac{d_{0,i}}{s_{0,i}} M_i$ and $\overline{P}_i = \frac{d_{0,i}d_{1,i}}{s_{0,i}s_{1,i}} P_i$, which have values between 0 and 1, and report this scale change in the definition of $k_{\text{on},i}$ and $k_{\text{off},i}$ (see section 1.1 of Additional file 1 for details). In the remainder of this article, the new variables will still be denoted by M_i and P_i as no confusion arises. The resulting normalized model is:

$$\begin{cases} E_i(t) : 0 \xrightarrow{k_{\text{on},i}} 1, \quad 1 \xrightarrow{k_{\text{off},i}} 0 \\ M_i'(t) = d_{0,i}\,(E_i(t) - M_i(t)) \\ P_i'(t) = d_{1,i}\,(M_i(t) - P_i(t)) \end{cases} \tag{4}$$

still omitting the dependence of $k_{\text{on},i}$ and $k_{\text{off},i}$ on $(P_1(t), \ldots, P_n(t))$ for clarity. This form enlightens the fact that $s_{0,i}$ and $s_{1,i}$ are just scaling constants: given a path $(E_i, M_i, P_i)_i$ of system (4), one can go back to the physical path by simply multiplying M_i by $(s_{0,i}/d_{0,i})$ and P_i by $(s_{0,i}/d_{0,i}) \times (s_{1,i}/d_{1,i})$.

Therefore, we get a general network model where each link between two genes is directed and has an explicit biochemical interpretation in terms of molecular interactions. The previous example is very simplistic but one can use virtually any model of chromatin dynamics to derive a form for $k_{\text{on},i}$ and $k_{\text{off},i}$, involving hit-and-run reactions, sequential binding, etc. [41]. Such aspects are still far from being completely understood [42–45] and this simple network model can hopefully be used to assess biological hypotheses. In the next part, we will introduce a more sophisticated interaction form based on an underlying probabilistic model, which is both "statistics-friendly" and interpretable as a non-equilibrium steady state of chromatin environment [43].

Some known mathematical results

Thanks to some recent theoretical results [40, 46], simple sufficient conditions on $k_{\text{on},i}$ and $k_{\text{off},i}$ ensure that the PDMP network model (4) is actually well-defined and that the overall joint distribution of $(E_i, M_i, P_i)_i$ converges as $t \to +\infty$ to a unique stationary distribution, which will be the basis of our statistical approach. Namely, we assume in this article that $k_{\text{on},i}$ and $k_{\text{off},i}$ are continuous functions of $(P_1, \ldots, P_n)$ and that they are greater than some positive constants. These conditions are satisfied in most interesting cases, including the above toy example (3) when $\theta_{i,0} > 0$.

Contrary to creation rates $s_{0,i}$ and $s_{1,i}$, degradation rates $d_{0,i}$ and $d_{1,i}$ play a crucial role in the dynamics of the system. Intuitively, the ratios $(k_{\text{on},i} + k_{\text{off},i})/d_{0,i}$ and $d_{0,i}/d_{1,i}$ respectively control the buffering of promoter noise by mRNA and the buffering of mRNA noise by proteins. A common situation is when promoter and mRNA dynamics are fast compared to proteins, i.e. when $d_{0,i} \gg d_{1,i}$ with $(k_{\text{on},i} + k_{\text{off},i})/d_{0,i}$ fixed. At the limit, the promoter-mRNA noise is fully averaged by proteins and model (4) simplifies into a deterministic system [47]:

$$P_i'(t) = d_{1,i}\left(\frac{k_{\text{on},i}(\mathbf{P}(t))}{k_{\text{on},i}(\mathbf{P}(t)) + k_{\text{off},i}(\mathbf{P}(t))} - P_i(t)\right) \tag{5}$$

where $\mathbf{P}(t) = (P_1(t), \ldots, P_n(t))$. The diffusion limit, which keeps a residual noise, can also be rigorously derived [48]. Unsurprisingly, one recovers the traditional

way of modelling gene regulatory networks with Hill-type interaction functions. Equation 5 is useful to get an insight into the behaviour of the system (4) for given $k_{\text{on},i}$ and $k_{\text{off},i}$, yet it should be used with caution. Indeed, the $d_{0,i}/d_{1,i}$ ratio has been shown to span a high range, averaging out to the value $d_{0,i}/d_{1,i} \approx 5$ in mammalian cells [31], for which taking the limit $d_{0,i} \gg d_{1,i}$ is not obvious. This is consistent with recent single-cell experiments showing a high variability of both mRNA and protein levels between cells [37]. In that sense, the PDMP model is much more robust than its deterministic/diffusion counterpart while keeping a similar level of mathematical complexity, which motivates our approach.

Simulation

We propose a simple algorithm to compute sample paths of our stochastic network model (4). It consists in a hybrid version of a basic ODE solver, making it efficient enough to perform massive simulations on large scale networks involving arbitrary numbers of molecules, which would be intractable with a classic molecule-based model (Fig. 3). The deterministic part of the algorithm is a standard explicit Euler scheme, while the stochastic part is based on the transient promoter distribution for single genes: this can be justified by the fact that during a small enough time interval, proteins remain almost constant so genes behave as if $k_{\text{on},i}$ and $k_{\text{off},i}$ were constant. We therefore use Bernoulli steps, in a similar way of a diffusion being simulated using gaussian steps.

After discretizing time with step δt, the numerical scheme is as follows. Starting from an initial state $\left(E_i^0, M_i^0, P_i^0\right)_i$, the update of the system from t to $t+\delta t$ is given by:

$$\begin{cases} E_i^{t+\delta t} \sim \mathcal{B}\left(\pi_i^t\right) \\ M_i^{t+\delta t} = (1 - d_{0,i}\delta t) M_i^t + d_{0,i}\delta t E_i^t \\ P_i^{t+\delta t} = (1 - d_{1,i}\delta t) P_i^t + d_{1,i}\delta t M_i^t \end{cases} \tag{6}$$

where the Bernoulli distribution parameter π_i^t is derived by locally solving the master equation for the promoter [49], i.e.

$$\pi_i^t = \frac{a_i^t}{a_i^t + b_i^t} + \left(E_i^t - \frac{a_i^t}{a_i^t + b_i^t}\right) e^{-(a_i^t + b_i^t)\delta t}$$

with the notation $a_i^t = k_{\text{on},i}(P_1^t, \ldots, P_n^t)$ and $b_i^t = k_{\text{off},i}(P_1^t, \ldots, P_n^t)$. Intuitively, the algorithm is valid when $\delta t \ll 1/\max_i \left\{K_{\text{on},i}, K_{\text{off},i}, d_{0,i}, d_{1,i}\right\}$ where $K_{\text{on},i}$ and $K_{\text{off},i}$ denote the maximum values of functions $k_{\text{on},i}$ and $k_{\text{off},i}$.

Deriving a tractable statistical model

We will now adopt a statistical perspective in order to deal with gene network inference, considering a set of observed cells. If they are evolving in the same environment for a long enough time, we can reasonably assume that their mRNA and protein levels follow the stationary distribution of an underlying gene network: this distribution can be used as a statistical likelihood for the cells. Furthermore assuming no cell-cell interactions (which may of course depend on the experimental context), we obtain a standard statistical problem with independent samples. Since the stationary distribution of the stochastic network model (4) is well-defined but a priori not analytically tractable, we will derive an explicit approximation and then reduce our inference problem to a traditional likelihood-based estimation. We will do so in two cases: when there is no self-interaction, and for a specific form of auto-activation.

Separating mRNA and protein timescales

It is for the moment very rare to experimentally obtain the amount of proteins for many genes at the single-cell level. We will therefore assume here that only mRNAs are observed. To deal with this problem, we take the protein timescale as our reference by fixing $d_{1,i}$ and assume that promoter dynamics are faster than proteins, i.e. $(k_{\text{on},i} +$

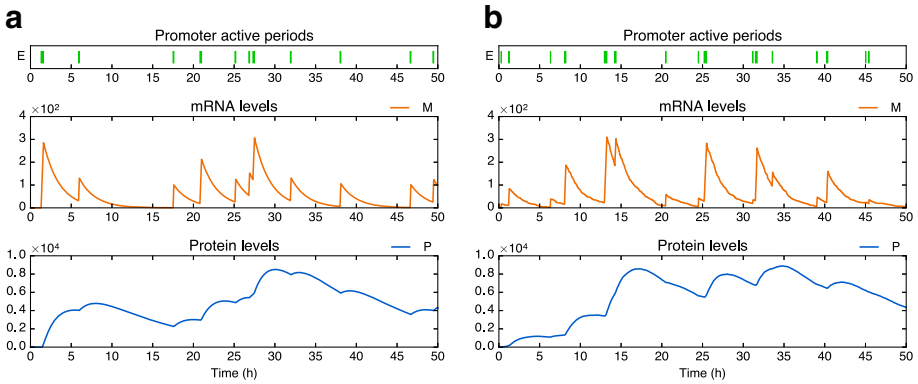


Fig. 3 Simulations of the two-state model for a single gene. **a** Sample path of the PDMP model using our hybrid numerical scheme (computation time ≈ 0.05 s). **b** Sample path of the classic model using exact stochastic simulation [27] (computation time ≈ 10 s). Parameters values are $k_{\text{on}} = 0.34$, $k_{\text{off}} = 10$, $s_0 = 10^3$, $s_1 = 10$, $d_0 = 0.5$ and $d_1 = 0.1$ (in h^{-1})

$k_{\text{off},i}) \gg d_{1,i}$ in a biologically relevant way, say $(k_{\text{on},i} + k_{\text{off},i})/d_{1,i} > 10$ (thus the deterministic limit (5) does not necessarily hold). Furthermore, in line with several recent experiments [37, 50], we assume that $d_{0,i}$ is sufficiently larger than $d_{1,i}$ so that the correlation between mRNAs and proteins produced by the gene is very small: model (4) then can be reduced by removing mRNA and making proteins directly depend on the promoters (see section 1.2 of Additional file 1). The result is

$$\begin{cases} E_i(t) : 0 \xrightarrow{k_{\text{on},i}} 1, \quad 1 \xrightarrow{k_{\text{off},i}} 0 \\ P_i'(t) = d_{1,i}\,(E_i(t) - P_i(t)) \end{cases} \tag{7}$$

which still admits the deterministic limit (5). Since mRNA dynamics are faster than proteins, one can also assume that, given protein levels $\mathbf{P} = (P_1, \ldots, P_n)$, each mRNA level M_i follows the quasi-steady state distribution

$$M_i \mid \mathbf{P} \sim \text{Beta}\left(\frac{k_{\text{on},i}(\mathbf{P})}{d_{0,i}}, \frac{k_{\text{off},i}(\mathbf{P})}{d_{0,i}}\right) \tag{8}$$

corresponding to the single-gene model [28, 39] with constant parameters $k_{\text{on},i}(\mathbf{P})$ and $k_{\text{off},i}(\mathbf{P})$. Numerically, this approximation works well even for moderate values of $d_{0,i}$, such as $d_{0,i} = 5 \times d_{1,i}$ (see the "Results" section).

Biologically, Eqs. (7) and (8) suggest that correlations between mRNA levels may not directly arise from correlations between promoters *states* (which in fact are weak because of $(k_{\text{on},i} + k_{\text{off},i}) \gg d_{1,i}$), but rather originate from correlations between promoter *parameters* $k_{\text{on},i}$ and $k_{\text{off},i}$, which themselves depend on the protein joint distribution.

Table 2 sums up the successive modelling steps introduced so far. From now on, we will always assume the form (8) for the mRNA distribution, and thus our model is reduced to Eq. (7) which only involves proteins.

Table 2 Successive dynamical models introduced in this article. We recall for each step the main feature and the form of the mRNA stationary distribution. The full network model (step 3) is used for simulations, while the simplified one (step 4) is used to derive the approximate statistical likelihood

1	*Single-gene, discrete* [29]	
	◇ All molecules are discrete	
	◇ mRNA distribution: Beta-Poisson	↓ Abundant species treated continuously
2	*Single-gene, PDMP* (1)	
	◇ Only the promoter is discrete	
	◇ mRNA distribution: Beta	↓ Introduction of interactions via k_{on}, k_{off}
3	*Network* (2), *normalized version* (4)	
	◇ Both accurate and fast to simulate	
	◇ mRNA distribution: unknown	↓ Timescale separation of Protein/mRNA ($d_0 \gg d_1$)
4	*Simplified network* (7)	
	◇ mRNA is removed from the network	
	◇ Conditional mRNA distribution: Beta (8)	

Hartree approximation

In this section, we present the Hartree approximation principle and provide an explicit formula in the particular case of no self-interaction. The simplified model (7) is still not analytically tractable, but it is now appropriate for employing the *self-consistent proteomic field* approximation introduced in [51, 52] and successfully applied in [53, 54]. More precisely, we will use its natural PDMP counterpart, which will be referred to as "Hartree approximation" since the main idea is similar to the Hartree approximation in physics [51]. It consists in assuming that genes behave as if they were independent from each other, but submitted to a common "proteomic field" created by all other genes. In other words, we transform the original problem of dimension 2^n into n independent problems of dimension 2 that are much easier to solve (see section 2 of Additional file 1 for details).

When $k_{\text{on},i}$ and $k_{\text{off},i}$ do not depend on P_i (i.e. no self-interaction), this approach results in approximating the protein stationary distribution of model (7) by the function

$$u(y) = \prod_{i=1}^{n} \frac{y_i^{a_i(y)-1}(1-y_i)^{b_i(y)-1}}{\text{B}(a_i(y), b_i(y))} \tag{9}$$

where $y = (y_1, \ldots, y_n) = (P_1, \ldots, P_n) = \mathbf{P}$, $a_i(y) = k_{\text{on},i}(y)/d_{1,i}$, $b_i(y) = k_{\text{off},i}(y)/d_{1,i}$ and B is the standard Beta function. Note that promoter states have been integrated out since they are not required by Eq. (8).

The function u is a heuristic approximation of a probability density function. It is only valid when interactions are not too strong, that is, when $k_{\text{on},i}$ and $k_{\text{off},i}$ are close enough to constants, and it becomes exact when they are true constants. Besides, it does not integrate to 1 in general. However, this approximation turns out to be very robust in practice and it has the great advantage to be fully explicit (and significantly simpler than in the non-PDMP case), thus providing a promising base for a statistical model.

When $k_{\text{on},i}$ and $k_{\text{off},i}$ depend on P_i, one can still explicitly compute the Hartree approximation in many cases: we will give an example in the next section. Alternatively, it is always possible to use formula (9) even with self-interactions, giving a correct approximation when the feedback is not too strong, as for other proteins.

An explicit form for interactions

We now propose an explicit definition of functions $k_{\text{on},i}$ and $k_{\text{off},i}$. Recent work [36, 55, 56] showed that apparent increased transcription actually reflects an increase in burst frequency rather than amplitude. We therefore decided to model only $k_{\text{on},i}$ as an actual function and to

keep $k_{\text{off},i}$ constant. In this view, the activation frequency of a gene can be influenced by ambiant proteins, whereas the active periods have a random duration that is dictated only by an intrinsic stability constant of the transcription machinery.

Our approach uses a description of the molecular activity around the promoter in a very similar way as Coulon et al. [42]. Accordingly, we make a quasi-steady state assumption to obtain $k_{\text{on},i}$. This idea based on thermodynamics was also used in the DREAM3 in-Silico Challenge [57] to simulate gene networks. However, only mean transcription rate was described (instead of promoter activity in our work), which is inappropriate to model bursty mRNA dynamics at the single-cell level.

We herein derive $k_{\text{on},i}$ from an underlying stochastic model for chromatin dynamics. We first introduce a set of abstract chromatin states, each state being associated with one of two possible rates of promoter activation, either a low rate $k_{0,i}$ or a high rate $k_{1,i} \gg k_{0,i}$. More specifically, such chromatin states may be envisioned as a coarse-grained description of the chromatin-associated parameters that are critical for transcription of gene i. Second, we assume a separation of timescales between the abstract chromatin model and the promoter activity, so that the promoter activation reaction depends only on the quasi-steady state of chromatin. In other words, the effective $k_{\text{on},i}$ is a combination of $k_{0,i}$ and $k_{1,i}$ which integrates all the chromatin states: its value depends on the probability of each state and a fortiori on the transitions between them. We propose a transition scheme which leads to an explicit form for $k_{\text{on},i}$, based on the idea that proteins can alter chromatin by hit-and-run reactions and potentially introduce a memory component. Some proteins thereby tend to stabilize it either in a "permissive" configuration (with rate $k_{1,i}$) or in a "non-permissive" configuration (with rate $k_{0,i}$), providing notions of *activation* and *inhibition*. A more precise definition and details of the derivation are provided in section 3 of Additional file 1.

The final form is the following. First, we define a function of every protein but P_i,

$$\Phi_i(y) = \exp(\theta_{i,i}) \prod_{j \neq i} \frac{1 + \exp(\theta_{i,j})(y_j/s_{i,j})^{m_{i,j}}}{1 + (y_j/s_{i,j})^{m_{i,j}}}$$

which may represent the external input of gene i. Then, $k_{\text{on},i}$ is defined by

$$k_{\text{on},i}(y) = \frac{k_{0,i} + k_{1,i}\Phi_i(y)(y_i/s_{i,i})^{m_{i,i}}}{1 + \Phi_i(y)(y_i/s_{i,i})^{m_{i,i}}}. \tag{10}$$

Hence, when the input $\Phi_i(y)$ is fixed, $k_{\text{on},i}$ is a standard Hill function which describes how gene i is self-activating, depending on the Hill coefficient $m_{i,i}$ (Fig. 4). The neutral value is set to $\Phi_i(y) = 1$, so that for this particular value, $s_{i,i}$ is the usual dissociation constant. Moreover, if $\theta_{i,j} = 0$ for all $j \neq i$, then Φ_i becomes the constant function $\Phi_i(y) = \exp(\theta_{i,i})$, and thus $\theta_{i,i}$ may be seen as a "basal" parameter, summing up all potential hidden inputs. On the contrary, if some $\theta_{i,j} > 0$ (resp. $\theta_{i,j} < 0$), then Φ_i becomes itself an increasing (resp. decreasing) Hill-type function of protein P_j, where $m_{i,j}$ and $s_{i,j}$ again play their usual roles.

The $n \times n$ matrix $\theta = (\theta_{i,j})$ therefore plays the same role as the interaction matrix in traditional network inference frameworks [8]. For $i \neq j$, $\theta_{i,j}$ quantifies the regulation of gene i by gene j (activation if $\theta_{i,j} > 0$, inhibition if $\theta_{i,j} < 0$, no influence if $\theta_{i,j} = 0$), and the diagonal term $\theta_{i,i}$ aggregates the "basal input" and the "self-activation strength" of gene i. Note that self-inhibition could be considered instead, but the choice has to be made before the inference since the self-interaction form is notoriously difficult to identify, especially in the stationary regime. In the remainder of this article, we assume that parameters $k_{0,i}$, $k_{1,i}$, $m_{i,j}$ and $s_{i,j}$ are known and we focus on inferring the matrix θ.

A benefit of the interaction form (10) is to allow for a fully explicit Hartree approximation of the protein distribution (see section 3 of Additional file 1 for details). In particular, if $m_{i,i} > 0$ and $c_i = (k_{1,i} - k_{0,i})/(d_{1,i}m_{i,i})$ is a

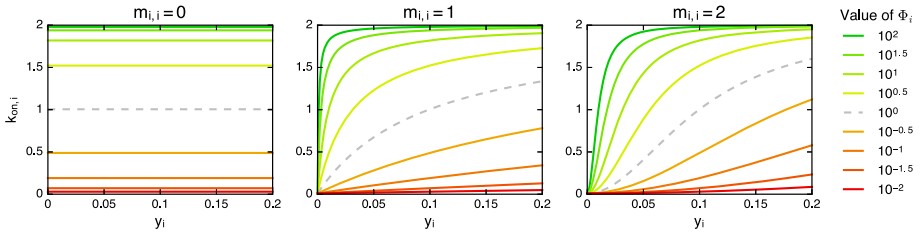


Fig. 4 Different auto-activation types in the network model. Each color corresponds to a fixed value of Φ_i in formula (10), and each curve represents $k_{\text{on},i}$ as a function of y_i for $m_{i,i} = 0$ (no feedback), $m_{i,i} = 1$ (monomer-type feedback) and $m_{i,i} = 2$ (dimer-type feedback). The neutral value $\Phi_i = 1$ is represented by a dashed gray line. Here $k_{0,i} = 0.01$, $k_{1,i} = 2$ and $s_{i,i} = 0.1$

positive integer for all i, the approximation is given by

$$u(y) = \prod_{i=1}^{n} \left(\sum_{r=0}^{c_i} p_{i,r}(y) \frac{y_i^{a_{i,r}-1}(1-y_i)^{b_i-1}}{\mathrm{B}(a_{i,r}, b_i)} \right) \tag{11}$$

with $a_{i,r} = ((c_i - r)k_{0,i} + rk_{1,i})/(d_{1,i}c_i)$, $b_i = k_{\mathrm{off},i}/d_{1,i}$ and

$$p_{i,r}(y) = \frac{\binom{c_i}{r} \mathrm{B}(a_{i,r}, b_i)(\Phi_i(y)/s_{i,i}^{m_{i,i}})^r}{\sum_{r'=0}^{c_i} \binom{c_i}{r'} \mathrm{B}(a_{i,r'}, b_i)(\Phi_i(y)/s_{i,i}^{m_{i,i}})^{r'}}.$$

In other words, the Hartree approximation (11) is a product of gene-specific distributions which are themselves mixtures of Beta distributions: for gene i, the $a_{i,r}$ correspond to "frequency modes" ranging from $k_{0,i}$ to $k_{1,i}$, weighted by the probabilities $p_{i,r}(y)$. It is straightforward to check that inhibitors tend to select the low burst frequencies of their target ($a_{i,r} \approx k_{0,i}$) while activators select the high frequencies ($a_{i,r} \approx k_{1,i}$). If $m_{i,i} = 0$ for some i, then $k_{\mathrm{on},i}$ does not depend on P_i so one just has to replace the i-th term in the product (11) with the single Beta form as in Eq. (9), which is equivalent to taking the limit $c_i \to +\infty$. Finally, when $m_{i,i} > 0$ but c_i is not an integer, using $\lceil c_i \rceil$ instead keeps a satisfying accuracy.

The statistical model in practice

Our statistical framework simply consists in combining the timescale separation (8) and the Hartree approximation (11) into a standard hidden Markov model. Indeed, conditionally to the proteins, mRNAs are independent and follow well-defined Beta distributions

$$\nu(x, y) = \prod_{i=1}^{n} \frac{x_i^{\widetilde{a}_i(y)-1}(1-x_i)^{\widetilde{b}_i(y)-1}}{\mathrm{B}(\widetilde{a}_i(y), \widetilde{b}_i(y))} \tag{12}$$

where $x = (x_1, \ldots, x_n) = (M_1, \ldots, M_n) = \mathbf{M}$, $\widetilde{a}_i(y) = k_{\mathrm{on},i}(y)/d_{0,i}$ and $\widetilde{b}_i(y) = k_{\mathrm{off},i}(y)/d_{0,i}$. Then one can use (11) to approximate the joint distribution of proteins. Hence, recalling the unknown interaction matrix θ, the inference problem for m cells with respective levels $(\mathbf{M}_k, \mathbf{P}_k)_{1 \leqslant k \leqslant m}$ is based on the (approximate) complete log-likelihood:

$$\begin{aligned} \ell &= \ell(\mathbf{M}_1, \ldots, \mathbf{M}_m, \mathbf{P}_1, \ldots, \mathbf{P}_m | \theta) \\ &= \sum_{k=1}^{m} \log(u(\mathbf{P}_k)) + \log(\nu(\mathbf{M}_k, \mathbf{P}_k)) \end{aligned} \tag{13}$$

where we used conditional factorization and independence of the cells.

The basic statistical inference problem would be to maximize the marginal likelihood of mRNA with respect to θ. Since this likelihood has no simple form, a typical way to perform inference is to use an Expectation-Maximization (EM) algorithm on the complete likelihood (13). However, the algorithm may be slow in practice because of the computation of expectations over proteins. A faster procedure consists in simplifying these expectations using the distribution modes: the resulting algorithm is often called "hard EM" or "classification EM" and is used in the "Results" section. Moreover, it is possible to encode some potential knowledge or constraints on the network by introducing a prior distribution $w(\theta)$. In this case, from Baye's rule, one can perform *maximum a posteriori* (MAP) estimation of θ by using the same EM algorithm but adding the penalization term $\log(w(\theta))$ to ℓ during the Maximization step (see section 4 of Additional file 1 and the "Results" section). Alternatively, a full bayesian approach, i.e. sampling from the posterior distribution of θ conditionally to $(\mathbf{M}_1, \ldots, \mathbf{M}_m)$, may also be considered using standard MCMC methods.

Taking advantage of the latent structure of proteins, we can also deal with missing data in a natural way: if the mRNA measurement of gene i is invalid in a cell k owing to technical problems, it is possible to ignore it by removing the i-th term in the conditional distribution of mRNAs (12). This only modifies the definition of ν for cell k in Eq. (13), ensuring that all valid data is effectively used for each cell.

Results

In this part, we first compare the distribution of the mechanistic model (4) to the mRNA quasi-steady state combined with Hartree approximation for proteins, on a simple toggle-switch example. Then, we show that the single-gene model with auto-activation can fit marginal mRNA distributions from real data better than the constant-k_{on} model. Finally, we successfully apply the inference procedure to various two-gene networks simulated from the mechanistic model.

Relevance of the approximate likelihood

Starting from the normalized mechanistic model (4), two approximations were used to derive the final statistical likelihood (13): the quasi-steady state assumption for mRNAs given protein levels, and the Hartree approximation for the joint distribution of proteins. Crucially, this approximate likelihood has to be close enough to the exact one in order to preserve the equivalence between inferring a network and fitting the mechanistic model. To get an idea of the accuracy, we considered a basic two-gene toggle switch defined by $k_{\mathrm{on},i}$ following Eq. (10) with the interaction matrix given by $\theta_{1,1} = \theta_{2,2} = 4$ and $\theta_{1,2} = \theta_{2,1} = -8$ (full parameter list in section 6 of Additional file 1). By computing sample paths (Fig. 5), we estimated the stationary distribution and compared it with our approximation, which appeared to be very satisfying, both for proteins and mRNAs (Fig. 6).

Fitting marginal mRNA distributions from real data

A particularity of single-cell data is to often exhibit bursty regimes for mRNA (meaning $k_{\mathrm{on}} \ll k_{\mathrm{off}}$ and $d_0 \ll k_{\mathrm{off}}$) and potentially also for proteins (adding $d_1 \ll k_{\mathrm{off}}$), which

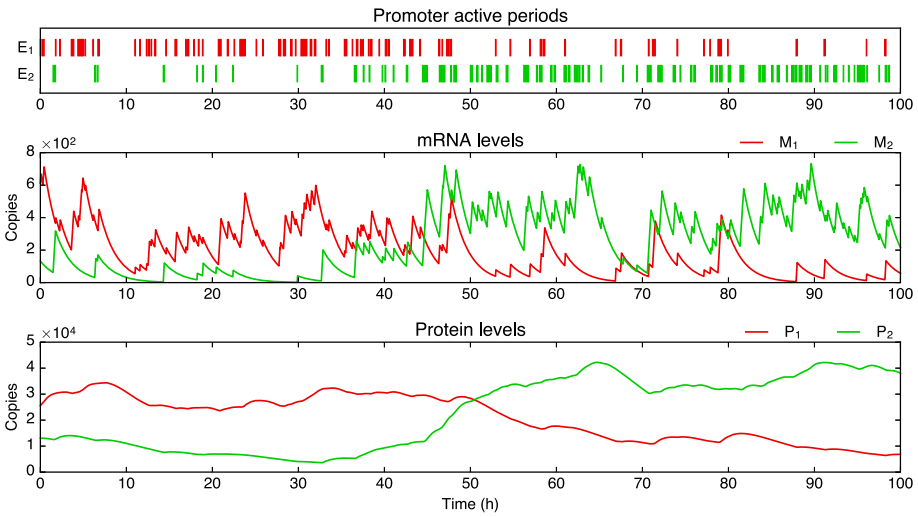


Fig. 5 Sample path of a two-gene toggle switch. The first gene is plotted in red and the second in green. While always staying in a bursty regime regarding mRNAs, genes can switch between high and low frequency modes (here at $t \approx 50$ h). From this example, it is clear that the overall joint distribution can contain correlations even if the bursts themselves are not coordinated

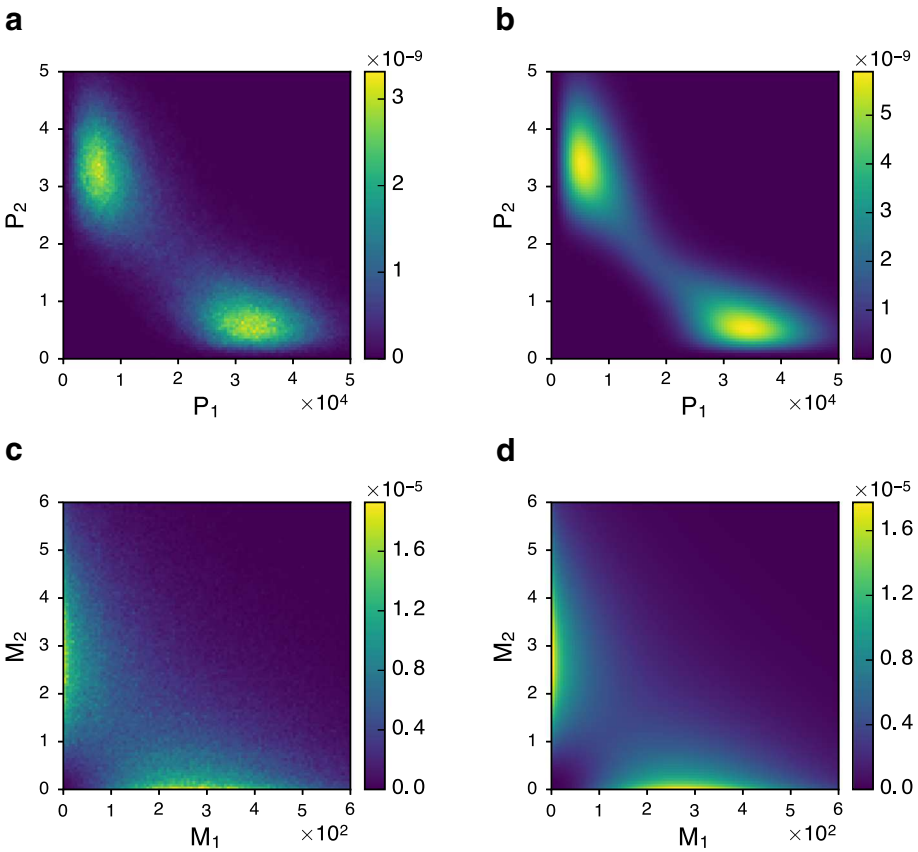


Fig. 6 Exact and approximate stationary distributions for the example of toggle switch. True distributions (left side) were estimated by sample path simulation, while approximations (right side) have explicit formulas. **a** True distribution of proteins. **b** Approximate distribution of proteins, from formula (11). **c** True distribution of mRNAs. **d** Approximate distribution of mRNAs, obtained by integrating the conditional distribution of mRNA (12) against (**b**)

are well fitted by Gamma distributions [37]. At this stage, it is worth mentioning that the Gamma distribution can be seen as a limit case of the Beta distribution. Intuitively, when $b \gg 1$ and $b \gg a$ (typically $a = k_{on}/d_0$ and $b = k_{off}/d_0$), most of the mass of the distribution Beta(a, b) is located at $x \ll 1$ so we have the first order approximation

$$x^{a-1}(1-x)^{b-1} = x^{a-1}\exp((b-1)\log(1-x)) \approx x^{a-1}\exp(-bx)$$

and thus Beta$(a, b) \approx \gamma(a, b)$. This way, formulas (11) and (12) can be easily transformed into Gamma-based distributions. Parameters s_0 and k_{off} then aggregate in k_{off}/s_0 because of the scaling property of the Gamma distribution, so only this ratio has to be inferred: from an applied perspective, it simply represents a scale parameter for each gene. This remark leads to a possible preprocessing phase that can be used for estimating the crucial basal parameters of the network, without requiring the knowledge of such scale parameters (see section 5 of Additional file 1).

In addition, our network model is able to generate multiple modes while keeping such bursty regimes (Fig. 5), as noticeable in the stationary distribution (11). Interestingly, this feature has already been considered in the literature by empirically introducing mixture distributions [58, 59]. As a first step toward applications, we compared our model in the simplest case (independent genes with auto-activation) to marginal distributions of single-cell mRNA measurements from [38]. Our model was fitted and compared to the basic two-state model in the bursty regime, i.e. to a simple Gamma distribution: Fig. 7 shows the example of the LDHA gene. Although very close when viewed in raw molecule numbers, the distributions differ after applying the transformation $x \mapsto x^\alpha$ with $\alpha = 1/3$, which tends to compress great values while preserving small values. The data becomes bimodal, suggesting the presence of two bursting regimes, a "normal" one and a very small "inhibited" one: the auto-activation model then performs better than the simple Gamma, which necessarily stays unimodal for $0 < \alpha < 1$. Note that the RTqPCR protocol used in [38] was shown to be far more sensitive than single-cell RNA-seq in the detection of low abundance transcripts [60]. Since the data also contains small nonzero values, this tends to support a true biological origin for the peak in zero. Besides, the case of distributions that are not bimodal until transformed also arises for proteins [61].

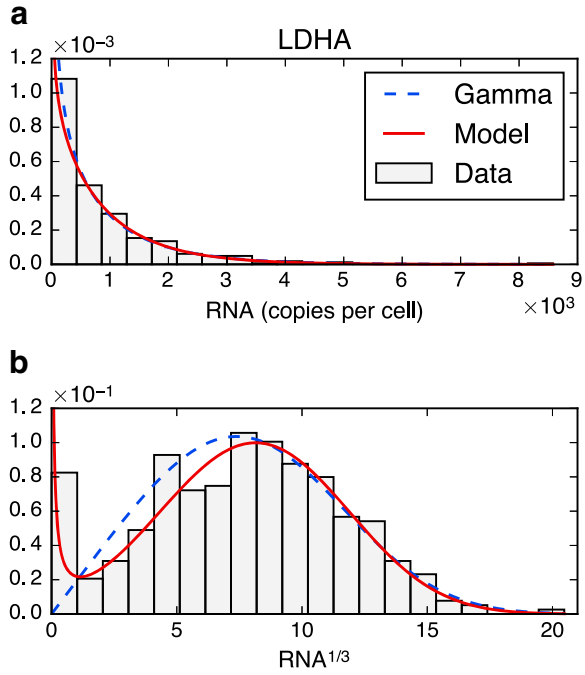


Fig. 7 Fitting marginal distributions from real single-cell data: example of the LDHA gene. The red curve is the stationary distribution associated with our interaction form (here a single gene with auto-activation), while the dashed blue curve corresponds to the basic two-state model in the bursty regime (Gamma distribution). **a** The raw data seems to be well fitted by the Gamma distribution, which in this view is close to our model. **b** Same fit viewed after applying the transformation $x \mapsto x^{1/3}$. The data becomes bimodal and the fit appears to be better with the auto-activation model

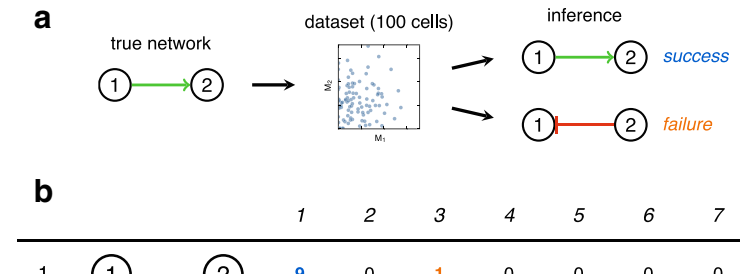
Application of the inference procedure

By construction of the mechanistic model, the interaction matrix θ can describe any oriented graph by explicitly defining causal quantitative links between genes, which is difficult to do within traditional statistical frameworks (e.g. bayesian networks or undirected Markov random fields). The logical downside is that identifiability issues seem inevitable. In a first attempt to assess this aspect, we implemented the inference method presented above and tested it on various two-gene networks, assuming auto-activation for each gene (i.e. $m_{i,i} > 0$) with Eq. (10) to maximize variability without considering perturbations of the system (parameter list in section 6 of Additional file 1).

We decided to investigate the worst case scenario in terms of cell numbers. We are fully aware of the existence of technologies allowing to interrogate thousands of cells simultaneously, but most of the recent studies still rely upon a much smaller number of cells. For each network, we therefore simulated mRNA snapshot data for 100 cells using the full PDMP model (4). We then inferred the matrix θ using a "hard EM" algorithm based on the likelihood (13), that is, alternatively maximizing the likelihood with respect to θ and with respect to the (unknown) protein levels of each cell. A lasso-like penalization term, corresponding to a prior distribution, was added to the $\theta_{i,j}$ for $i \neq j$ to obtain true zeros – so that the inferred network topology is clear – and to prevent keeping both $\theta_{i,j}$

and $\theta_{j,i}$ when one is significantly weaker (see section 4 of Additional file 1 for details of the penalization and the whole procedure).

We obtained highly encouraging results since every structure was inferred with a high probability of success (Fig. 8), meaning that the non-diagonal (i.e. interaction) terms of θ had the right sign and were nonzero at the right places. A list of the inferred values is provided in Additional file 1: Table S3. It is very important at that stage to emphasize that we are not trying to infer θ exactly: we only assess whether it has a zero or nonzero value and its sign. Although the results tend to support the identifiability of the full matrix θ in this simple two-gene case, one has to be aware that the quantity we maximize (an approximate likelihood) is a priori non convex and can have several local maxima (i.e. networks that are relevant



b

		1	2	3	4	5	6	7
1	1 ⟲ 2	9	0	1	0	0	0	0
2	1 → 2	2	8	0	0	0	0	0
3	1 ← 2	1	0	7	2	0	0	0
4	1 ⇄ 2	0	0	0	10	0	0	0
5	1 ⊣ 2	2	0	0	0	7	0	1
6	1 ⊣ 2	1	0	0	0	0	8	1
7	1 ⟲ 2	0	0	0	0	0	0	10

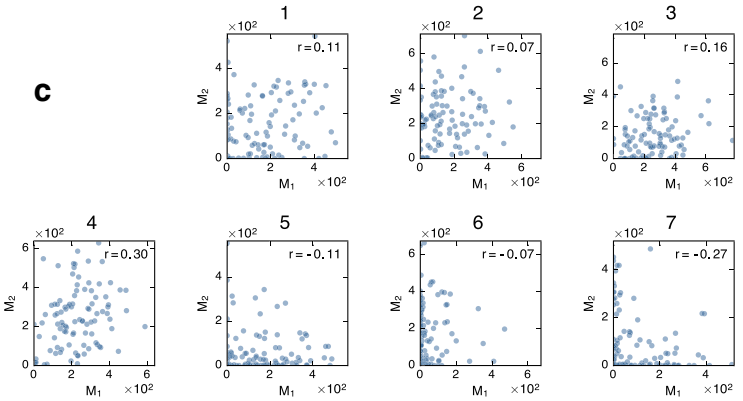


Fig. 8 Testing our inference method on simple networks. **a** For each network, numbered from 1 to 7, we simulated 100 cells using the full mechanistic model until the stationary regime was reached. Then we took a snapshot of their mRNA levels and inferred the parameters from this data. The result was called successful when the inferred structure (topology and nature of the links) was the same as the true network. **b** For each network (rows), 10 datasets were simulated and the results were reported by counting the number of inferred θ corresponding to each structure (columns), highlighting successes (blue) and failures (orange). The perfect inference would lead to 10 for all the diagonal terms and 0 everywhere else. **c** Examples of simulated mRNA datasets (one for each network). Although having coherent signs, Pearson's correlation coefficients (top right of each plot) would clearly be insufficient to distinguish between the different networks

candidates to explain the data). The result of the inference thus can depend on the starting point: in this first approach we chose the null matrix to be the starting point for θ, which corresponds to the – biologically relevant – expectation of "balanced" behaviors (e.g. we do not expect $\theta_{1,1} \ll \theta_{2,2}$). Alternatively, one can consider some probabilistic prior knowledge on θ to implement a (possibly rough) idea of parameter values from a Bayesian viewpoint: it is worth mentioning that any knockout information can be implemented this way in our model.

Finally, we assessed the inference behavior in the presence of dropouts, i.e. genes expressed at a low level in a cell that give rise to zeros after measurement [4]. Our first tests tend to indicate that our approach is robust regarding dropouts, in the sense that up to 30% of simulated dropouts does not drastically affect the estimation of θ once the other parameters have been estimated correctly (see Additional file 1: Table S4 for an example).

Discussion

In this paper, we introduce a general stochastic model for gene regulatory networks, which can describe bursty gene expression as observed in individual cells. Instead of using ordinary differential equations, for which cells would structurally all behave the same way, we adopt a more detailed point of view including stochasticity as a fundamental component through the two-state promoter model. This model is but a simplification of the complexity of the real molecular processes [42]. Modifications have been proposed, from the existence of a refractory period [23] to its attenuation by nuclear buffering [62]. In bacteria, the two states originate from the accumulation of positive supercoiling on DNA which stops transcription [63]. In eukaryotes, although its molecular basis is not quite understood, the two-state model is a remarkable compromise between simplicity and the ability to capture real-life data [18, 22, 36–38]. Our PDMP framework appears to be conceptually very similar to the *random dynamical system* proposed in [64] but it has two major advantages: time does not have to be discretized, and the mathematical analysis is significantly easier. We also note that a similar framework appears in [65, 66] and that a closely related PDMP – which can be seen as the limit of our model for infinitely short bursts – has recently been described in [67].

We then derive an explicit approximation of the stationary distribution and propose to use it as a statistical likelihood to infer networks from single-cell data. The main ingredient is the separation of three physical timescales – chromatin, promoter/RNA, and proteins – and the core idea is to use the self consistent proteomic field approximation from [51, 52] in a slightly simpler mathematical framework, providing fully explicit formulas that make possible the massive computations usually needed for parameter inference. From this viewpoint, it is a rather simple approach and we hope it can be adapted or improved in more specific contexts, for example in the study of lineage commitment [68]. Besides, the main framework does not necessarily has to include an underlying chromatin model and thus it can in principle also be used to describe gene networks in procaryotes.

Mechanistic modelling and statistical inference

An important quality of the PDMP network model is that the simulation algorithm is comparable in speed with classic ODE and diffusion systems, while providing an effective approximation of the "perfect", fully discrete, molecular counterpart [33, 35]. It is worth noticing that the PDMP – at least the promoter-mRNA system – naturally appears as an example of Poisson representation [28, 69], that is, not a simple approximation but rather the core component of the *exact* distribution of the discrete molecular model. Furthermore, such a simulation speed allowed us to compare our approximate likelihood with the true likelihood for a simple two-gene toggle switch, giving excellent results (Fig. 6). This obviously does not constitute a proof of robustness for every network: a proper quantitative (theoretical or numeric) comparison is beyond the scope of this article but would be extremely valuable. Intuitively, it should work for any number of genes, provided that interactions are not too strong.

Besides, some widely used ODE frameworks [8, 17, 57] can be seen as the fast-promoter limit of the PDMP model: this limit may not always hold in practice, especially in the bursty regime. In particular, Fig. 5 highlights the risk of using mRNA levels as a proxy for protein levels. It also explains why ordering single-cell mRNA measurements by pseudo-time may not always be relevant, as found in [38]. In [70], the authors use a hybrid model of gene expression to infer regulatory networks: it is very close to the diffusion limit of our reduced model (7) with the difference that the discrete component, called "promoter" by the authors, would correspond to the "frequency mode" in the present article, as visible for proteins in Fig. 5. From such a perspective, our approach adds a description of bursty mRNA dynamics that allows for fitting single-cell data such as in Fig. 7.

Finally, our method performed well for simple two-gene networks (Fig. 8), showing that part of the causal information remains present in the stationary distribution: this suggests that it is indeed possible to retrieve network structures with a mechanistic interpretation, even from bursty mRNA data.

Perspectives

We focused here on presenting the key ideas behind the general network model and the inference method: the logical next step is to apply it to real data and with a larger

number of genes, which is the subject of work in progress in our group. In particular, we propose a functional preprocessing phase, detailed in section 5 of Additional file 1, that only requires the knowledge of the ratio $d_{0,i}/d_{1,i}$ to estimate all the relevant parameters before inferring θ. The ratio between protein and mRNA degradation rates (or half-lives) hence appears to be the minimum required for such a mechanistic approach to be relevant. Depending upon the species, mRNA and protein half-lives values can be found in the literature (see e.g. [31] for human proteins half-lives), or should be estimated from ad hoc experiments.

From a computational point of view, the main challenge is the algorithmic complexity induced by the fact that proteins are not observed and have to be treated as latent variables. There is a priori no possibility of reducing this without loosing too much accuracy, and therefore some finely optimized algorithms may be required to make the method scalable. Furthermore, the identifiability properties of the interaction matrix θ seem difficult to derive theoretically. In this paper we focused on the stationary distribution for simplicity: importantly, several aspects such as time dependence (computing the Hartree approximation in transitory regime) or perturbations (changing the cell's medium or performing knockouts [71], which can be naturally embedded in our framework) could greatly improve the practical identifiability.

From a biological point of view, our model does not really describe individual cells but rather a concatenation of trajectories obtained by following cells throughout divisions. Experiments suggest that it should be a relevant approximation, providing one considers mRNA and proteins levels in terms of concentrations instead of molecule numbers [72], which is made possible by the PDMP framework. In this view, the cell cycle results in increasing the apparent degradation rates – because of the increase in cell volume followed by division – and thus plays a crucial role for very stable proteins. However, at such a description level, many aspects of possible compensation mechanisms [73] and chromatin dynamics [74] remain to be elucidated. Regarding the latter aspect, our abstract chromatin states were not modeled from real-life data – chromatin composition for instance – but our approach is relevant in that partitioning into dual-type chromatin states as we did is now known as a pervasive feature of all eukaryotic genomes [75–78].

Conclusions

Protein and mRNA measurements in individual cells have revealed the importance of stochasticity in gene expression, which may potentially affect many aspects of gene regulation within cells. The traditional paradigm of gene network dynamics consisting in a deterministic structure plus an external noise – historically based on population-averaged data – should therefore be questioned, as such a noise appears to be itself part of the network structure and far from a small perturbation.

By modelling gene networks using piecewise-deterministic Markov processes, which are a simple way to introduce the minimum amount of mechanistic, non-diffusive stochasticity (corresponding to low molecule numbers), we derived a likelihood-based statistical model with interpretable parameters that successfully describes single-cell expression data. Our first results show that oriented interactions can indeed be inferred using such a method. Hence, this type of approach may take gene network inference to the next level by optimally exploiting single-cell data and improving the physical interpretability of inferred networks.

Additional file

Additional file 1: Additional file 1. Supplementary information. This document contains details of the theoretical derivations and all the parameter values used in the examples. (PDF 362 kb)

Acknowledgements
We thank Geneviève Fourel (LBMC/ENSL) for critical reading of the manuscript and Anne-Laure Fougères (ICJ) for her constant support. We also thank all members of the SBDM and Dracula teams for enlightening discussions, and BioSyL Federation and Ecofect Labex for inspiring scientific events.

Funding
This work was supported by funding from the Institut Rhônalpin des Systèmes Complexes (IXXI) and from the French agency ANR (ICEBERG; ANR-IABI-3096).

Availability of data and material
The data used to obtain Fig. 7 is available from [38]. The inference method was implemented in Scilab and the code is available upon request.

Authors' contributions
UH, AB, TE and OG designed the study. UH performed the theoretical derivations, implemented the algorithms and conceived/analyzed the examples. UH, AB, TE and OG interpreted the results and wrote the paper. All authors read and approved the final manuscript.

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Publisher's Note

Author details
[1]Univ Lyon, ENS de Lyon, Univ Claude Bernard, CNRS UMR 5239, INSERM U1210, Laboratory of Biology and Modelling of the Cell, 46 allée d'Italie Site Jacques Monod, F-69007 Lyon, France. [2]Inria Team Dracula, Inria Center Grenoble Rhône-Alpes, Lyon, France. [3]Univ Lyon, Université Claude Bernard Lyon 1, CNRS UMR 5208, Institut Camille Jordan, 43 blvd. du 11 novembre 1918, F-6962 Villeurbanne Cedex, France. [4]The CoSMo company, 5 passage du Vercors, 69007 Lyon, France.

Received: 12 May 2017 Accepted: 9 November 2017
Published online: 21 November 2017

References

1. Hecker M, Lambeck S, Toepfer S, van Someren E, Guthke R. Gene regulatory network inference: data integration in dynamic models-a review. BioSystems. 2009;96(1):86–103.
2. Kanter I, Kalisky T. Single cell transcriptomics: methods and applications. Front Oncol. 2015;5:53.
3. Tang F, Barbacioru C, Wang Y, Nordman E, Lee C, Xu N, Wang X, Bodeau J, Tuch BB, Siddiqui A, Lao K, Surani MA. mRNA-Seq whole-transcriptome analysis of a single cell. Nat Methods. 2009;6(5):377–82.
4. Wagner A, Regev A, Yosef N. Revealing the vectors of cellular identity with single-cell genomics. Nat Biotechnol. 2016;34(11):1145–1160.
5. Huang S. Non-genetic heterogeneity of cells in development: more than just noise. Development. 2009;136(23):3853–3862.
6. Eldar A, Elowitz MB. Functional roles for noise in genetic circuits. Nature. 2010;467(7312):167–173.
7. Dueck H, Eberwine J, Kim J. Variation is function: Are single cell differences functionally important? Bioessays. 2015;38:172–180.
8. Mizeranschi A, Zheng H, Thompson P, Dubitzky W. Evaluating a common semi-mechanistic mathematical model of gene-regulatory networks. BMC Syst Biol. 2015;9(5):1–12.
9. Matsumoto H, Kiryu H, Furusawa C, Ko MSH, Ko SBH, Gouda N, Hayashi T, Nikaido I. Scode: An efficient regulatory network inference algorithm from single-cell rna-seq during differentiation. Bioinformatics. 2017;33(15):2314–2321.
10. Symmons O, Raj A. What's luck got to do with it: Single cells, multiple fates, and biological nondeterminism. Mol Cell. 2016;62(5):788–802.
11. Munsky B, Trinh B, Khammash M. Listening to the noise: random fluctuations reveal gene network parameters. Mol Syst Biol. 2009;5(1):1–7.
12. Zimmer C, Sahle S, Pahle J. Exploiting intrinsic fluctuations to identify model parameters. IET Syst Biol. 2015;9(2):64–73.
13. Cai X, Bazerque JA, Giannakis GB. Inference of gene regulatory networks with sparse structural equation models exploiting genetic perturbations. PLoS Comput Biol. 2013;9(5):1–13.
14. Djordjevic D, Yang A, Zadoorian A, Rungrugeecharoen K, Ho JWK. How difficult is inference of mammalian causal gene regulatory networks? PLoS One. 2014;9(11):1–10.
15. Angulo MT, Moreno JA, Lippner G, Barabási A-L, Liu Y-Y. Fundamental limitations of network reconstruction from temporal data. J R Soc Interface. 2017;14(127):1–6.
16. Moignard V, Woodhouse S, Haghverdi L, Lilly AJ, Tanaka Y, Wilkinson AC, Buettner F, Macaulay IC, Jawaid W, Diamanti E, Nishikawa S-I, Piterman N, Kouskoff V, Theis FJ, Fisher J, Göttgens B. Decoding the regulatory network of early blood development from single-cell gene expression measurements. Nat Biotechnol. 2015;33(3):1–8.
17. Ocone A, Haghverdi L, Mueller NS, Theis FJ. Reconstructing gene regulatory dynamics from high-dimensional single-cell snapshot data. Bioinformatics. 2015;31(12):89–86.
18. Kim JK, Marioni JC. Inferring the kinetics of stochastic gene expression from single-cell RNA-sequencing data. Genome Biol. 2013;14:7.
19. Marbach D, Costello JC, Küffner R, Vega NM, Prill RJ, Camacho DM, Allison KR, DREAM5 Consortium, Kellis M, Collins JJ, Stolovitzky G. Wisdom of crowds for robust gene network inference. Nat Methods. 2012;9(8):796–804.
20. Raser JM, O'Shea EK. Control of stochasticity in eukaryotic gene expression. Science. 2004;304(5678):1811–1814.
21. Becskei A, Kaufmann BB, van Oudenaarden A. Contributions of low molecule number and chromosomal positioning to stochastic gene expression. Nat Genet. 2005;37(9):937–944.
22. Raj A, Peskin CS, Tranchina D, Vargas DY, Tyagi S. Stochastic mRNA Synthesis in Mammalian Cells. PLoS Biology. 2006;4(10):1707–1719.
23. Suter DM, Molina N, Gatfield D, Schneider K, Schibler U, Naef F. Mammalian genes are transcribed with widely different bursting kinetics. Science. 2011;332(6028):472–474.
24. Ko MSH. A stochastic model for gene induction. J Theor Biol. 1991;153: 181–194.
25. Ko MSH, Nakauchi H, Takahashi N. The dose dependence of glucocorticoid-inducible gene expression results from changes in the number of transcriptionally active templates. EMBO J. 1990;9(9): 2835–2842.
26. Larson DR. What do expression dynamics tell us about the mechanism of transcription? Curr Opin Genet Dev. 2011;21(5):591–599.
27. Gillespie DT. Exact stochastic simulation of coupled chemical reactions. J Phys Chem. 1977;81(25):2340–2361.
28. Dattani J, Barahona M. Stochastic models of gene transcription with upstream drives: exact solution and sample path characterization. J R Soc Interface. 2017;14(126):1–20.
29. Shahrezaei V, Swain PS. Analytical distributions for stochastic gene expression. PNAS. 2008;105(45):17256–17261.
30. Iyer-Biswas S, Hayot F, Jayaprakash C. Stochasticity of gene products from transcriptional pulsing. Phys Rev E Stat Nonlin Soft Matter Phys. 2009;79:1–9.
31. Schwanhäusser B, Busse D, Li N, Dittmar G, Schuchhardt J, Wolf J, Chen W, Selbach M. Global quantification of mammalian gene expression control. Nature. 2011;495:337–342.
32. Davis MHA. Piecewise-deterministic markov processes: A general class of non-diffusion stochastic models. J R Stat Soc. 1984;46(3):353–388.
33. Crudu A, Debussche A, Radulescu O. Hybrid stochastic simplifications for multiscale gene networks. BMC Syst Biol. 2009;3(1):89.
34. Crudu A, Debussche A, Muller A, Radulescu O. Convergence of stochastic gene networks to hybrid piecewise deterministic processes. Ann Appl Probab. 2012;22(5):1822–1859.
35. Lin YT, Galla T. Bursting noise in gene expression dynamics: linking microscopic and mesoscopic models. J R Soc Interface. 2016;13:1–11.
36. Viñuelas J, Kaneko G, Coulon A, Vallin E, Morin V, Mejia-Pous C, Kupiec J-J, Beslon G, Gandrillon O. Quantifying the contribution of chromatin dynamics to stochastic gene expression reveals long, locus-dependent periods between transcriptional bursts. BMC Biol. 2013;11(1):15.
37. Albayrak C, Jordi CA, Zechner C, Lin J, Bichsel CA, Khammash M, Tay S. Digital quantification of proteins and mrna in single mammalian cells. Mol Cell. 2016;61:914–924.
38. Richard A, Boullu L, Herbach U, Bonnafoux A, Morin V, Vallin E, Guillemin A, Papili Gao N, Gunawan R, Cosette J, Arnaud O, Kupiec J-J, Espinasse T, Gonin-Giraud S, Gandrillon O. Single-cell-based analysis highlights a surge in cell-to-cell molecular variability preceding irreversible commitment in a differentiation process. PLoS Biol. 2016;14(12):1–35.
39. Boxma O, Kaspi H, Kella O, Perry D. On/Off Storage Systems with State-Dependent Input, Output, and Switching Rates. Probab Eng Inf Sci. 2005;19:1–14.
40. Benaïm M, Le Borgne S, Malrieu F, Zitt P-A. Quantitative ergodicity for some switched dynamical systems. Electron Commun Probab. 2012;17(56):1–14.
41. Ong KM, Blackford, JA Jr, Kagan BL, Simons, SS Jr, Chow CC. A theoretical framework for gene induction and experimental comparisons. PNAS. 2010;107(15):7107–7112.
42. Coulon A, Gandrillon O, Beslon G. On the spontaneous stochastic dynamics of a single gene: complexity of the molecular interplay at the promoter. BMC Syst Biol. 2010;4:2.
43. Coulon A, Chow CC, Singer RH, Larson DR. Eukaryotic transcriptional dynamics: from single molecules to cell populations. Nat Rev Genet. 2013;14(8):1–13.
44. Friedman N, Rando OJ. Epigenomics and the structure of the living genome. Genome Res. 2015;25(10):1482–1490.
45. Bintu L, Yong J, Antebi YE, McCue K, Kazuki Y, Uno N, Oshimura M, Elowitz MB. Dynamics of epigenetic regulation at the single-cell level. Science. 2016;351(6274):720–724.
46. Benaïm M, Le Borgne S, Malrieu F, Zitt P-A. Qualitative properties of certain piecewise deterministic Markov processes. Annales de l'Institut Henri Poincaré - Probabilités et Statistiques. 2015;51(3):1040–1075.
47. Faggionato A, Gabrielli D, Crivellari MR. Non-equilibrium thermodynamics of piecewise deterministic markov processes. J Stat Phys. 2009;137:259–304.
48. Pakdaman K, Thieullen M, Wainrib G. Asymptotic expansion and central limit theorem for multiscale piecewise-deterministic Markov processes. Stoch Process Appl. 2012;122:2292–2318.
49. Peccoud J, Ycart B. Markovian Modelling of Gene Product Synthesis. Theor Popul Biol. 1995;48:222–234.
50. Li G-W, Xie XS. Central dogma at the single-molecule level in living cells. Nature. 2011;475(7356):308–315.
51. Sasai M, Wolynes PG. Stochastic gene expression as a many-body problem. PNAS. 2003;100(5):2374–2379.

52. Walczak AM, Sasai M, Wolynes PG. Self-consistent proteomic field theory of stochastic gene switches. Biophys J. 2005;88:828–850.
53. Kim K-Y, Wang J. Potential energy landscape and robustness of a gene regulatory network: toggle switch. PLoS Comput Biol. 2007;3(3):565–577.
54. Zhang B, Wolynes PG. Stem cell differentiation as a many-body problem. PNAS. 2014;111(28):10185–10190.
55. Senecal A, Munsky B, Proux F, Ly N, Braye FE, Zimmer C, Mueller F, Darzacq X. Transcription Factors Modulate c-Fos Transcriptional Bursts. Cell Rep. 2014;8:75–83.
56. Fukaya T, Lim B, Levine M. Enhancer control of transcriptional bursting. Cell. 2016;166(2):358–368.
57. Marbach D, Prill RJ, Schaffter T, Mattiussi C, Floreano D, Stolovitzky G. Revealing strengths and weaknesses of methods for gene network inference. PNAS. 2010;107(14):6286–6291.
58. Gu J, Gu Q, Wang X, Yu P, Lin W. Sphinx: modeling transcriptional heterogeneity in single-cell RNA-Seq. bioRxiv preprint. 2015.
59. Ghazanfar S, Bisogni AJ, Ormerod JT, Lin DM, Yang JYH. Integrated single cell data analysis reveals cell specific networks and novel coactivation markers. BMC Syst Biol. 2016;10:127.
60. Mojtahedi M, Skupin A, Zhou J, Castano IG, Leong-Quong RYY, Chang HH, Giuliani A, Huang S. Cell fate decision as high-dimensional critical state transition. PLOS Biol. 2016;14(12):1–28.
61. Sokolik C, Liu Y, Bauer D, McPherson J, Broeker M, Heimberg G, Qi LS, Sivak DA, Thomson M. Transcription factor competition allows embryonic stem cells to distinguish authentic signals from noise. Cell Syst. 2015;1:117–129.
62. Battich N, Stoeger T, Pelkmans L. Control of transcript variability in single mammalian cells. Cell. 2015;163(7):1596–1610.
63. Chong S, Chen C, Ge H, Xie XS. Mechanism of transcriptional bursting in bacteria. Cell. 2014;158(2):314–326.
64. Antoneli F, Ferreira RC, Briones MRS. A model of gene expression based on random dynamical systems reveals modularity properties of gene regulatory networks. Math Biosci. 2016;276:82–100.
65. Potoyan DA, Wolynes PG. Dichotomous noise models of gene switches. J Chem Phys. 2015;143(19):195101.
66. Hufton PG, Lin YT, Galla T, McKane AJ. Intrinsic noise in systems with switching environments. Phys Rev E. 2016;93(5):052119.
67. Pájaro M, Alonso AA, Otero-Muras I, Vázquez C. Stochastic modeling and numerical simulation of gene regulatory networks with protein bursting. J Theor Biol. 2017;421:51–70.
68. Teles J, Pina C, Edén P, Ohlsson M, Enver T, Peterson C. Transcriptional Regulation of Lineage Commitment - A Stochastic Model of Cell Fate Decisions. PLoS Comput Biol. 2013;9(8):1–13.
69. Schnoerr D, Grima R, Sanguinetti G. Cox process representation and inference for stochastic reaction-diffusion processes. Nat Commun. 2016;7:1–11.
70. Ocone A, Millar AJ, Sanguinetti G. Hybrid regulatory models: a statistically tractable approach to model regulatory network dynamics. Bioinformatics. 2013;29(7):910–916.
71. Pinna A, Soranzo N, de la Fuente A. From knockouts to networks: establishing direct cause-effect relationships through graph analysis. PLoS One. 2010;5(10):1–8.
72. Corre G, Stockholm D, Arnaud O, Kaneko G, Viñuelas J, Yamagata Y, Neildez-Nguyen TMA, Kupiec J-J, Beslon G, Gandrillon O, Paldi A. Stochastic Fluctuations and Distributed Control of Gene Expression Impact Cellular Memory. PLoS ONE. 2014;9(12):115574.
73. Padovan-Merhar O, Nair GP, Biaesch AG, Mayer A, Scarfone S, Foley SW, Wu AR, Churchman LS, Singh A, Raj A. Single mammalian cells compensate for differences in cellular volume and dna copy number through independent global transcriptional mechanisms. Mol Cell. 2015;58(2):339–352.
74. Hathaway NA, Bell O, Hodges C, Miller EL, Neel DS, Crabtree GR. Dynamics and memory of heterochromatin in living cells. Cell. 2012;149(7):1447–1460.
75. Fourel G, Magdinier F, Gilson E. Insulator dynamics and the setting of chromatin domains. BioEssays. 2004;26(5):523–532.
76. Kueng S, Oppikofer M, Gasser SM. Sir proteins and the assembly of silent chromatin in budding yeast. Annu Rev Genet. 2013;47:275–306.
77. Rao SSP, Huntley MH, Durand NC, Stamenova EK, Bochkov ID, Robinson JT, Sanborn AL, Machol I, Omer AD, Lander ES, Aiden EL. A 3d map of the human genome at kilobase resolution reveals principles of chromatin looping. Cell. 2014;159(7):1665–1680.
78. Obersriebnig MJ, Pallesen EMH, Sneppen K, Trusina A, Thon G. Nucleation and spreading of a heterochromatic domain in fission yeast. Nat Commun. 2016;7:1–11.

Appendix 6.A – EM algorithm for network inference

6.A.1 EM algorithm for MAP estimation

Here we briefly recall the formulation of the Expectation-Maximization (EM) algorithm for *maximum a posteriori* (MAP) estimation. Consider the probabilistic hierarchical model defined by the distribution of proteins $p(y|\theta)$, the distribution of mRNA given proteins $p(x|y,\theta)$, and a prior distribution $p(\theta)$ on the parameters. Assuming we only observe x, we want to infer θ by MAP estimation, that is, find a mode – hopefully the highest – of the posterior distribution $p(\theta \,|\, x)$, which satisfies by Baye's rule :

$$p(\theta \,|\, x) = \int p(\theta, y \,|\, x) dy \quad \text{where} \quad p(\theta, y \,|\, x) = p(y \,|\, \theta) p(x \,|\, y, \theta) \frac{p(\theta)}{p(x)}.$$

As $p(\theta \,|\, x)$ has a too complex expression to be efficiently maximized, the EM algorithm rather uses $\ell_\theta(x, y) = \log(p(\theta, y \,|\, x))$ by iteratively computing $\theta^{t+1} = \arg\max_\theta \{Q(\theta, \theta^t)\}$ given θ^t, where

$$\theta \mapsto Q(\theta, \theta^t) = \int \ell_\theta(x, y) p(y \,|\, x, \theta^t) dy. \tag{6.1}$$

A well-known result states that at each step we in fact maximize a lower bound of $p(\theta \,|\, x)$, which is the key point of the algorithm and makes it a particular case of "variational method" (see Jordan *et al.*, 1999 for example). Now, since $p(x)$ (resp. $p(\theta)$) does not depend on θ (resp. y), it turns out that

$$\arg\max_\theta \{Q(\theta, \theta^t)\} = \arg\max_\theta \{\overline{Q}(\theta, \theta^t) - g(\theta)\}$$

where $g(\theta) = -\log(p(\theta))$ and $\overline{Q}(\theta, \theta^t) = \int [\log p(y \,|\, \theta) + \log p(x \,|\, y, \theta)] p(y \,|\, x, \theta^t) dy$ is the more standard quantity that appears in the "frequentist" EM algorithm for maximum likelihood estimation. Hence, considering a prior on θ simply results in adding a penalization term $g(\theta)$ during the M step in the algorithm.

For example, if we assume that $\theta_{i,j}$ for $i \neq j$ are independent and follow Laplace distributions, i.e. $p(\theta) = \prod_{i \neq j} \frac{\lambda}{2} \exp(-\lambda |\theta_{i,j}|)$, then $g(\theta) = \lambda \sum_{i \neq j} |\theta_{i,j}| + C$ where $C = n(n-1) \log(2/\lambda)$. Since C does not depend on θ, this is equivalent to the standard L^1 (lasso) penalization, which is well known to enforce the sparsity of the network.

6.A.2 Custom prior on the interactions

Here we consider a custom prior to deal with oriented interactions. Indeed, for every pair of nodes $\{i, j\}$ there are two possible interactions with respective parameters $\theta_{i,j}$ and $\theta_{j,i}$, but it is likely that only one is actually present in the true network. Hence, we want $\theta_{i,j}$ and $\theta_{j,i}$ to "compete" against each other so that only one is nonzero after MAP estimation, unless there is enough evidence in the data that both interactions are present. To this aim, we define the following prior :

$$p(\theta) \propto \exp\left(-\lambda \sum_{i \neq j} |\theta_{i,j}| - \lambda \alpha \sum_{i < j} |\theta_{i,j} \theta_{j,i}| \right) \tag{6.2}$$

with $\lambda, \alpha \geqslant 0$. Thus α can be seen as a competition parameter, the case $\alpha = 0$ leading to the standard lasso penalization parametrized by λ.

6.A.3 The algorithm in practice

As visible in (6.1), the true EM algorithm involves integration against the distribution $p(y \,|\, x, \theta)$, which does not allow for direct numerical integration because of the dimension ($y \in \mathbb{R}^n$). To overcome this problem, a first option is Monte Carlo integration – typically by MCMC – leading to a "stochastic EM" algorithm that is slow but accurate if samples are large enough. A faster option consists in approximating $p(y \,|\, x, \theta)$ by its highest mode, i.e. by the Dirac mass $\delta_{\widehat{y}}$ where $\widehat{y} = \arg\max_y \{p(y \,|\, x, \theta)\}$. Then it is worth noticing that since $p(y \,|\, x, \theta) \propto p(y \,|\, \theta) p(x \,|\, y, \theta)$, the whole procedure can be seen as performing a coordinate ascent on the function $(\theta, y) \mapsto p(\theta, y \,|\, x)$. We chose this option for the examples : it is sometimes called "hard" or "classification" EM, since a particular case leads to the well-known k-means clustering algorithm (Celeux et Govaert, 1991). Unfortunately, theoretical foundations of the true EM algorithm are lost by the hard EM (we do not maximize a lower bound of $p(\theta \,|\, x)$ anymore), but it often gives satisfying results while requiring much less computational time.

In practice, the procedure is the following. Suppose we observe mRNA levels in m independent cells, and let $\mathbf{x}_k \in \mathbb{R}^n$ (resp. $\mathbf{y}_k \in \mathbb{R}^n$) denote the mRNA (resp. protein) levels of cell k. In line with previous sections and letting $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_m)$ and $\mathbf{y} = (\mathbf{y}_1, \ldots, \mathbf{y}_m)$ for simplicity, we define the objective function

$$\mathcal{F}(\mathbf{y}, \theta) = \ell(\mathbf{x}, \mathbf{y}, \theta) - g(\theta) \tag{6.3}$$

where the complete log-likelihood $\ell(\mathbf{x}, \mathbf{y}, \theta)$ and the penalization $g(\theta)$ are given by

$$\ell(\mathbf{x}, \mathbf{y}, \theta) = \sum_{k=1}^{m} \log(u(\mathbf{y}_k, \theta)) + \log(v(\mathbf{x}_k, \mathbf{y}_k, \theta)) \quad \text{and} \quad g(\theta) = \lambda \sum_{i \neq j} |\theta_{i,j}| + \lambda\alpha \sum_{i<j} |\theta_{i,j}\theta_{j,i}|,$$

with $u(y, \theta) = p(y|\theta)$ and $v(x, y, \theta) = p(x|y, \theta)$.

The algorithm then simply consists in iterating the following two steps until convergence :

$$\mathbf{y}^{t+1} = \arg\max_{\mathbf{y}} \{\mathcal{F}(\mathbf{y}, \theta^t)\} \tag{6.4}$$

$$\theta^{t+1} = \arg\max_{\theta} \{\mathcal{F}(\mathbf{y}^{t+1}, \theta)\} \tag{6.5}$$

The "approximate E step" (6.4) can be performed using a standard gradient method since u and v are smooth functions of y. The "penalized M step" (6.5) is a non-smooth maximization problem since g is non-smooth, but it can be performed using a proximal gradient method detailed in the next section. The form of $\ell(\mathbf{x}, \mathbf{y}, \theta)$ is such that we just need to compute $\nabla \log u$ and $\nabla \log v$.

6.A.4 Proximal gradient method

Here we recall a standard proximal gradient method (Parikh et Boyd, 2013) to solve the M step (6.5) and provide the proximal operator associated with $g(\theta)$. Note that the method

seems to converge in practice, even if g is not convex. It is based on the update

$$\theta^{(k+1)} = \mathrm{prox}_\gamma\left(\theta^{(k)} + \gamma\nabla_\theta \ell(\mathbf{x}, \mathbf{y}, \theta^{(k)})\right)$$

where $\gamma > 0$ is a step size (learing rate) and prox_γ is the proximal operator associated with $g(\theta)$, defined on $\Theta \simeq \mathbb{R}^{n^2-n}$ by

$$\mathrm{prox}_\gamma(\tau) = \arg\min_{\theta\in\Theta}\left\{g(\theta) + \frac{1}{2\gamma}\sum_{i\neq j}(\theta_{i,j} - \tau_{i,j})^2\right\}.$$

In fact, for any $i, j \in \{1, \ldots, n\}$ such that $i \neq j$, one can see that $\theta_{i,j}$ and $\theta_{j,i}$ appear in the minimized quantity as independent of all other θ components. Hence, one just has to compute

$$\mathrm{prox}_\gamma(\tau_1, \tau_2) = \arg\min_{(\theta_1,\theta_2)\in\mathbb{R}^2}\left\{\lambda\left(|\theta_1| + |\theta_2| + \alpha|\theta_1\theta_2|\right) + \frac{1}{2\gamma}\left((\theta_1 - \tau_1)^2 + (\theta_2 - \tau_2)^2\right)\right\}$$

and use it for any $(\tau_1, \tau_2) = (\tau_{i,j}, \tau_{j,i}) \in \mathbb{R}^2$ to obtain the corresponding components of $\mathrm{prox}_\gamma(\tau)$. Then, letting $\varepsilon = \lambda\gamma$ and assuming γ small enough such that $\alpha\varepsilon < 1$, we obtain

$$\mathrm{prox}_\gamma(\tau_1, \tau_2) = \frac{1}{1 - (\alpha\varepsilon)^2}(h_1, h_2)$$

with 9 cases for the value of (h_1, h_2) depending on (τ_1, τ_2), given by :

1. $\left\{\begin{array}{l}\tau_1 > \varepsilon\\ \tau_1 > \varepsilon(1 + \alpha(\tau_2 - \varepsilon))\\ \tau_2 > \varepsilon(1 + \alpha(\tau_1 - \varepsilon))\end{array}\right. \Rightarrow \left\{\begin{array}{l}h_1 = \tau_1 - \varepsilon(1 + \alpha(\tau_2 - \varepsilon))\\ h_2 = \tau_2 - \varepsilon(1 + \alpha(\tau_1 - \varepsilon))\end{array}\right.$
2. $\left\{\begin{array}{l}\tau_1 > \varepsilon\\ |\tau_2| \leqslant \varepsilon(1 + \alpha(\tau_1 - \varepsilon))\end{array}\right. \Rightarrow \left\{\begin{array}{l}h_1 = \tau_1 - \varepsilon\\ h_2 = 0\end{array}\right.$
3. $\left\{\begin{array}{l}\tau_1 > \varepsilon\\ \tau_1 > \varepsilon(1 + \alpha(-\tau_2 - \varepsilon))\\ \tau_2 < -\varepsilon(1 + \alpha(\tau_1 - \varepsilon))\end{array}\right. \Rightarrow \left\{\begin{array}{l}h_1 = \tau_1 - \varepsilon(1 + \alpha(-\tau_2 - \varepsilon))\\ h_2 = \tau_2 + \varepsilon(1 + \alpha(\tau_1 - \varepsilon))\end{array}\right.$
4. $\left\{\begin{array}{l}|\tau_1| \leqslant \varepsilon(1 + \alpha(-\tau_2 - \varepsilon))\\ \tau_2 < -\varepsilon\end{array}\right. \Rightarrow \left\{\begin{array}{l}h_1 = 0\\ h_2 = \tau_2 + \varepsilon\end{array}\right.$
5. $\left\{\begin{array}{l}\tau_1 < -\varepsilon\\ \tau_1 < -\varepsilon(1 + \alpha(-\tau_2 - \varepsilon))\\ \tau_2 < -\varepsilon(1 + \alpha(-\tau_1 - \varepsilon))\end{array}\right. \Rightarrow \left\{\begin{array}{l}h_1 = \tau_1 + \varepsilon(1 + \alpha(-\tau_2 - \varepsilon))\\ h_2 = \tau_2 + \varepsilon(1 + \alpha(-\tau_1 - \varepsilon))\end{array}\right.$
6. $\left\{\begin{array}{l}\tau_1 < -\varepsilon\\ |\tau_2| \leqslant \varepsilon(1 + \alpha(-\tau_1 - \varepsilon))\end{array}\right. \Rightarrow \left\{\begin{array}{l}h_1 = \tau_1 + \varepsilon\\ h_2 = 0\end{array}\right.$
7. $\left\{\begin{array}{l}\tau_1 < -\varepsilon\\ \tau_1 < -\varepsilon(1 + \alpha(\tau_2 - \varepsilon))\\ \tau_2 > \varepsilon(1 + \alpha(-\tau_1 - \varepsilon))\end{array}\right. \Rightarrow \left\{\begin{array}{l}h_1 = \tau_1 + \varepsilon(1 + \alpha(\tau_2 - \varepsilon))\\ h_2 = \tau_2 - \varepsilon(1 + \alpha(-\tau_1 - \varepsilon))\end{array}\right.$
8. $\left\{\begin{array}{l}|\tau_1| \leqslant \varepsilon(1 + \alpha(\tau_2 - \varepsilon))\\ \tau_2 > \varepsilon\end{array}\right. \Rightarrow \left\{\begin{array}{l}h_1 = 0\\ h_2 = \tau_2 - \varepsilon\end{array}\right.$
9. $\left\{\begin{array}{l}|\tau_1| \leqslant \varepsilon\\ |\tau_2| \leqslant \varepsilon\end{array}\right. \Rightarrow \left\{\begin{array}{l}h_1 = 0\\ h_2 = 0\end{array}\right.$

These 9 cases form a partition of $\mathbb{R}^2$ and are represented in Figure 6.1. One can check that the case $\alpha = 0$ collapses to the usual proximal operator associated with lasso penalization.

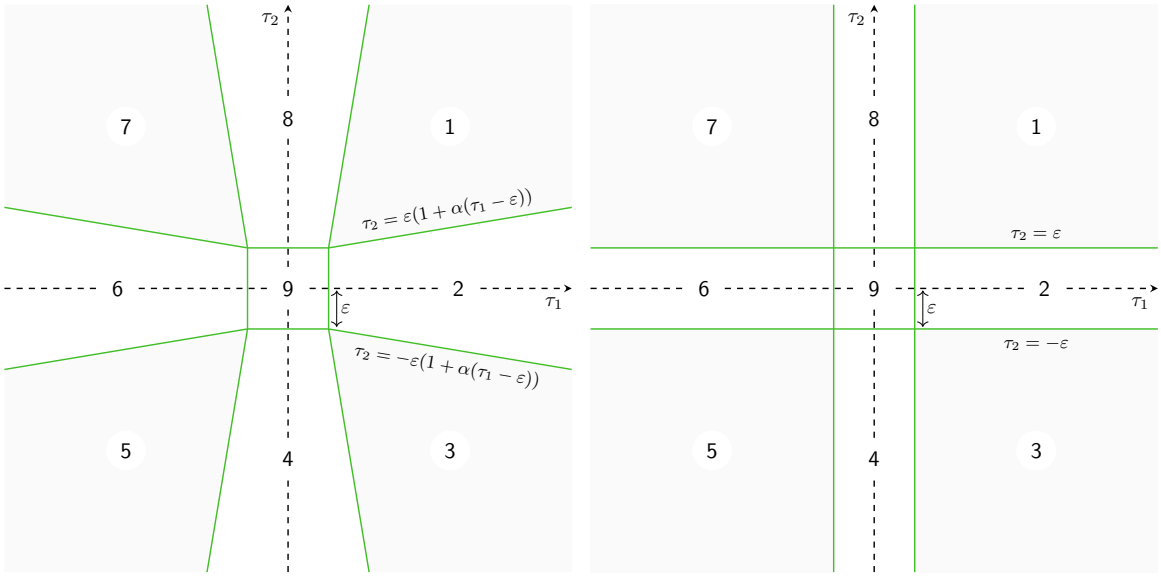


Figure 6.1 – Partition of $\mathbb{R}^2$ associated with the proximal operator, for $\alpha > 0$ (left) and $\alpha = 0$ (right). Gray areas correspond to a usual gradient and white areas correspond to a threshold.

To obtain the results of Figure 8 in the paper, we used $\lambda = 10$, $\alpha = 5$ and $\gamma = 10^{-4}$. In a broader context, one may use cross-validation to derive appropriate values for λ and α.

6.A.5 Estimating the hyperparameters

Here we describe a heuristic method to estimate the model-specific hyperparameters. Note that one should know at least the ratio $d_{0,i}/d_{1,i}$ (e.g. by measuring mRNA and protein half-lives). When it is unavailable, we propose to use the default value $d_{0,i}/d_{1,i} = 5$.

1. Estimate $\widetilde{a}_{0,i} = k_{0,i}/d_{0,i}$, $\widetilde{a}_{1,i} = k_{1,i}/d_{0,i}$, $\widetilde{b}_i = k_{\mathrm{off},i}/s_{0,i}$, $\widetilde{c}_i$ and Φ from the likelihood

$$f(x_i) \propto \sum_{r=0}^{\widetilde{c}_i} \Phi^r {x_i}^{(1-r/\widetilde{c}_i)\widetilde{a}_{0,i}+(r/\widetilde{c}_i)\widetilde{a}_{1,i}-1} e^{-\widetilde{b}_i x_i}.$$

   This can be done using an EM algorithm for each value of $\widetilde{c}_i$ in some range (e.g. $\widetilde{c}_i \in [\![1, 10]\!]$), and then choosing the "arg max" tuple $(\widetilde{a}_{0,i}, \widetilde{a}_{1,i}, \widetilde{b}_i, \widetilde{c}_i, \Phi)$. Afterwards, $\widetilde{a}_{0,i}$, $\widetilde{a}_{1,i}$, $\widetilde{b}_i$ and $\widetilde{c}_i$ are stored (Φ only serves this step).

2. Consistently with the definition of the model, we set $m_{i,i} = (\widetilde{a}_{1,i} - \widetilde{a}_{0,i})/\widetilde{c}_i$,

$$a_{0,i} = \frac{k_{0,i}}{d_{1,i}} = \frac{d_{0,i}}{d_{1,i}} \cdot \widetilde{a}_{0,i}, \quad a_{1,i} = \frac{k_{1,i}}{d_{1,i}} = \frac{d_{0,i}}{d_{1,i}} \cdot \widetilde{a}_{1,i}, \quad c_i = \frac{k_{1,i} - k_{0,i}}{d_{1,i} m_{i,i}} = \frac{d_{0,i}}{d_{1,i}} \cdot \widetilde{c}_i$$

   and we choose $b_i = \frac{d_{0,i}}{d_{1,i}} \cdot \widetilde{b}_i$. Then we define, as an approximation in the bursty regime,

$$s_{i,i} = \frac{1}{b_i} \left(\frac{\Gamma(a_{1,i})}{\Gamma(a_{0,i})} \right)^{1/(a_{1,i}-a_{0,i})}.$$

   Note that such b_i is not the "true" value as we would need to know $\frac{d_{1,i}}{s_{1,i}}$ to apply the formula $b_i = \frac{d_{1,i}}{s_{1,i}} \cdot \frac{d_{0,i}}{d_{1,i}} \cdot \widetilde{b}_i$. Fortunately, the network inference does not depend on this scale parameter since the Hill threshold $s_{i,i}$ is proportional to $1/b_i$.

Chapitre 7

Application de l'auto-modèle Gamma-Binomial

Dans ce dernier chapitre, on s'intéresse à l'application de l'auto-modèle Gamma-Binomial défini dans la section 2.3. On commence par présenter une méthode variationnelle pour l'inférence (Jordan *et al.*, 1999 ; Wainwright et Jordan, 2008), qui s'avère nettement plus rapide pour ce modèle qu'une méthode de type Monte Carlo. On présente ensuite un résultat encourageant pour un réseau simulé par le modèle PDMP, ce qui donne une certaine intuition sur la manière pertinente d'utiliser ce modèle statistique. Enfin, on montre quelques premiers résultats sur les données réelles correspondant au chapitre 4.

7.1 Méthode variationnelle

On considère le modèle de la Définition 2.22, i.e. pour $c_i \in \mathbb{N}^*$ et $a_{0,i}, a_{1,i}, a_{2,i} > 0$ fixés :

$$p_\theta(x,z) \propto \exp\left(\sum_{i=1}^n \theta_i z_i + a_{0,i}\ln x_i + a_{1,i} z_i \ln x_i - a_{2,i} x_i + \sum_{i<j} \theta_{i,j} z_i z_j\right) \prod_{i=1}^n {x_i}^{-1} \binom{c_i}{z_i}$$

où $x \in {\mathbb{R}_+^*}^n$ est observé (ARNm) et $z \in \prod_{i=1}^n [\![0, c_i]\!]$ est caché (modes de fréquences des promoteurs), et avec $\theta \in \mathcal{S}_n(\mathbb{R})$ le paramètre d'intérêt. Notre objectif est de maximiser $p_\theta(x)$ en θ, mais cette marginale est compliquée contrairement à $p_\theta(x,z)$. L'idée des méthodes variationnelles est d'utiliser le résultat suivant :

$$\begin{aligned} \ln p_\theta(x) = \ln \sum_z p_\theta(x,z) &= \ln \sum_z \frac{p_\theta(x,z)}{q(z)} q(z) \\ &\geqslant \sum_z \ln\left(\frac{p_\theta(x,z)}{q(z)}\right) q(z) \\ &= \sum_z (\ln p_\theta(x,z) - \ln q(z)) q(z) \end{aligned} \tag{7.1}$$

valable pour toute distribution $q(z)$ ne s'annulant pas, et qui est simplement une conséquence de l'inégalité de Jensen appliquée à la fonction concave $s \mapsto \ln(s)$. Noter que la différence entre le terme de gauche et le terme de droite peut s'interpréter comme la divergence de Kullback-Leibler $\mathrm{D}(q(\cdot) \| p_\theta(\cdot|x))$ comme souligné par Blei *et al.* (2003). L'idée fondamentale

de notre approche consiste à utiliser comme substitut de $p_\theta(z|x)$ la famille de lois :

$$q(z) = q_\alpha(z) \propto \prod_{i=1}^{n} \binom{c_i}{z_i} \exp(\alpha_i z_i)$$

où $\alpha \in \mathbb{R}^n$ est à choisir au mieux pour que $q_\alpha(z)$ soit le proche possible de $p_\theta(z|x)$. Noter qu'il y aura ainsi un vecteur α pour chaque observation x. En maximisant alternativement le membre de droite de (7.1) en α et en θ, on est donc sûr de maximiser une borne inférieure de $p_\theta(x)$. Le choix judicieux de $q_\alpha(z)$ permet de simplifier les $\binom{c_i}{z_i}$ et on obtient :

$$\begin{aligned}\ln p_\theta(x,z) - \ln q_\alpha(z) = &\sum_{i=1}^{n} \theta_i z_i + (a_{0,i}-1)\ln x_i + a_{1,i} z_i \ln x_i - a_{2,i} x_i \\ &+ \sum_{i<j} \theta_{i,j} z_i z_j \\ &+ \sum_{i=1}^{n} \ln A_{q,i}(\alpha_i) - \alpha_i z_i \\ &- \ln A_p(\theta)\end{aligned}$$

où $A_{q,i}(\alpha_i) = \sum_{z_i=0}^{c_i} \binom{c_i}{z_i} \exp(\alpha_i z_i) = (1+e^{\alpha_i})^{c_i}$ et $A_p(\theta)$ est la constante de normalisation de $p_\theta(x,z)$. Cela va nous permettre de définir une fonction objectif Q dépendant de θ et α.

7.1.1 Fonction objectif

On pose $Q(\theta,\alpha) = \sum_z (\ln p_\theta(x,z) - \ln q_\alpha(z)) q_\alpha(z) - C$ où C représente tous les termes qui n'ont pas d'impact dans l'inférence car constants en (θ,α). On a après simplifications :

$$Q(\theta,\alpha) = \sum_{i=1}^{n} (\theta_i + a_{1,i}\ln x_i - \alpha_i)\mathbb{E}_q[Z_i] + \ln A_{q,i}(\alpha_i) + \sum_{i<j} \theta_{i,j}\mathbb{E}_q[Z_i Z_j] - \ln A_p(\theta).$$

En exploitant le fait que $p_\theta(x,z)$ forme une famille exponentielle en θ, on obtient

$$\partial_{\theta_i} Q = \mathbb{E}_q[Z_i] - \mathbb{E}_p[Z_i] \quad \text{et} \quad \partial_{\theta_{i,j}} Q = \mathbb{E}_q[Z_i Z_j] - \mathbb{E}_p[Z_i Z_j] \tag{7.2}$$

où l'on a noté respectivement $\mathbb{E}_p$ et $\mathbb{E}_q$ les espérances sous les lois $p_\theta(z)$ et $q_\alpha(z)$. Notons que la forme particulière de q fournit

$$\mathbb{E}_q[Z_i] = \frac{c_i e^{\alpha_i}}{1+e^{\alpha_i}}, \quad \mathrm{Var}_q[Z_i] = \partial_{\alpha_i}\mathbb{E}_q[Z_i] = \frac{c_i e^{\alpha_i}}{(1+e^{\alpha_i})^2},$$

$$\mathbb{E}_q[Z_i Z_j] = \mathbb{E}_q[Z_i]\mathbb{E}_q[Z_j] = \frac{c_i e^{\alpha_i}}{1+e^{\alpha_i}} \frac{c_j e^{\alpha_j}}{1+e^{\alpha_j}}.$$

Remarque 7.1. Si l'on avait calculé le gradient de $\ell(\theta) = \ln p_\theta(x)$, on aurait obtenu

$$\partial_{\theta_i}\ell = \mathbb{E}_p[Z_i|X] - \mathbb{E}_p[Z_i] \quad \text{et} \quad \partial_{\theta_{i,j}}\ell = \mathbb{E}_p[Z_i Z_j|X] - \mathbb{E}_p[Z_i Z_j],$$

ce qui est tout à fait en accord avec le paradigme $q_\alpha(z) \approx p_\theta(z|x)$. En fait, une correspondance aussi explicite est typiquement liée aux familles exponentielles (Wainwright et Jordan, 2008).

Par ailleurs, connaissant θ, on obtient :

$$\partial_{\alpha_i} Q = \left(\theta_i + a_{1,i} \ln x_i - \alpha_i + \sum_{j \neq i} \theta_{i,j} \mathbb{E}_q[Z_j] \right) \mathrm{Var}_q[Z_i].$$

Or, comme $\mathrm{Var}_q[Z_i] > 0$ pour tout α_i, on peut annuler explicitement ce gradient en prenant :

$$\alpha_i = a_{1,i} \ln x_i + \theta_i + \sum_{j \neq i} \theta_{i,j} \frac{c_j e^{\alpha_j}}{1 + e^{\alpha_j}}. \tag{7.3}$$

Noter que cette équation fournit directement une méthode de point fixe pour annuler le gradient de Q en α : il suffit d'appliquer (7.3) en série pour tout $i \in [\![1, n]\!]$ jusqu'à convergence. Finalement, on obtient notre *méthode variationnelle* qui consiste simplement à itérer les deux étapes suivantes jusqu'à convergence :

1. Sachant θ, maximiser $\alpha \mapsto Q(\theta, \alpha)$ grâce à (7.3) puis mettre à jour α ;
2. Sachant α, maximiser $\theta \mapsto Q(\theta, \alpha)$ grâce à (7.2) puis mettre à jour θ.

La première étape peut se voir naturellement comme une généralisation de l'étape E (pour *Expectation*) de l'algorithme EM, ce qui place ce dernier comme un cas particulier de la classe des méthodes variationnelles. L'algorithme EM est en fait la méthode variationnelle de précision optimale (Jordan *et al.*, 1999) : il correspond au choix $q(z) = p_\theta(z|x)$, ce qui revient à réaliser à chaque itération l'égalité dans (7.1). Le fait d'avoir remplacé $p_\theta(z|x)$ par une loi $q_\alpha(z)$ plus simple permet d'accélérer grandement les calculs, surtout si l'on a beaucoup de données (il y a un α et un $p_\theta(z|x)$ spécifiques pour chaque observation, et α est bien plus rapide à calculer). L'avantage d'avoir développé le formalisme ainsi est que l'on a la garantie de maximiser une borne inférieure de $p_\theta(x)$.

Remarque 7.2. L'étape 1 est extrêmement rapide, ce qui fait tout l'intérêt de la méthode variationnelle par rapport au gradient classique où il faudrait calculer $\mathbb{E}_p[Z_i|X]$ et $\mathbb{E}_p[Z_i Z_j|X]$ pour chaque observation X. En fait, cela permet de d'inférer notre champ de Markov caché à une vitesse très proche de celle de la version classique où Z serait observé. L'étape 2 nécessite en revanche de calculer $\mathbb{E}_p[Z_i]$ et $\mathbb{E}_p[Z_i Z_j]$ à chaque mise à jour de θ. Elle est caractéristique des champs de Markov en général et c'est là que réside la difficulté de l'inférence. Le calcul exact des espérances peut être fait de manière intelligente avec l'algorithme *junction tree*, mais cela reste souvent trop lent. Nous avons choisi une alternative plus rapide et plus simple, l'algorithme *belief propagation* (Wainwright et Jordan, 2008 ; Murphy, 2012). Il s'agit d'une méthode exacte si le graphe associé à θ est un arbre, et approchée sinon. En pratique elle semble très satisfaisante dans notre cas. Noter que l'on peut aussi estimer $\mathbb{E}_p[Z_i]$ et $\mathbb{E}_p[Z_i Z_j]$ par Monte Carlo puisque la forme d'auto-modèle fournit naturellement un algorithme de Gibbs pour échantillonner asymptotiquement selon $\mathcal{L}(X, Z)$.

7.1.2 Cas de valeurs manquantes

S'il manque certains x_i (comme c'est effectivement le cas pour les données du chapitre 4), on aimerait redéfinir intelligemment $q_\alpha(z)$ de façon à garder le résultat théorique variationnel tout en étant capable de rapidement maximiser en α. En fait, si x_j est manquant, on intègre

d'abord $p_\theta(x,z)$ en x_j, ce qui donne

$$\begin{aligned}\ln \tilde{p}_\theta(x,z) = &\sum_{i\neq j} \ln \binom{c_i}{z_i} + \theta_i z_i + (a_{0,i}-1)\ln x_i + a_{1,i} z_i \ln x_i - a_{2,i} x_i \\ &+ \sum_{i<j} \theta_{i,j} z_i z_j - \ln A_p(\theta) \\ &+ \ln \binom{c_j}{z_j} + \theta_j z_j + \ln \Gamma(a_{0,j} + a_{1,j} z_j) - (a_{0,j} + a_{1,j} z_j)\ln(a_{2,j})\end{aligned}$$

On a alors une solution naturelle : on se base sur le cas d'un gène isolé et on essaie de calibrer la marginale de z_j. Cela amène à modifier q en transformant le terme $\binom{c_j}{z_j}\exp(\alpha_j z_j)$ en

$$\binom{c_j}{z_j}\exp(\alpha_j z_j)\Gamma(a_{0,j} + a_{1,j} z_j)$$

De manière cruciale, c'est encore une famille exponentielle en α. La formule de Q devient :

$$\begin{aligned}Q(\theta,\alpha) = &\sum_{i\neq j} (\theta_i + a_{1,i}\ln x_i - \alpha_i)\mathbb{E}_q[Z_i] + \ln A_{q,i}(\alpha_i) + \sum_{i<j} \theta_{i,j}\mathbb{E}_q[Z_i Z_j] - \ln A_p(\theta) \\ &+ (\theta_j + a_{1,j}\ln a_{2,j} - \alpha_j)\mathbb{E}_q[Z_j] + \ln \tilde{A}_{q,j}(\alpha_j)\end{aligned}$$

Les formules des dérivées $\partial_\theta Q$ restent les mêmes, la seule espérance qui change étant $\mathbb{E}_q[Z_j]$ qui est facilement calculable numériquement (et on garde l'indépendance, ce qui est la clé de la méthode). Et enfin concernant α, on peut encore factoriser par la variance et les mises à jours restent les mêmes. On obtient finalement la règle générale de mise à jour :

$$\text{Si } x_i \text{ est observé}, \quad \alpha_i = \theta_i + a_{1,i}\ln x_i + \sum_{j\neq i} \theta_{i,j}\mathbb{E}_q[Z_j]$$

$$\text{Si } x_i \text{ n'est pas observé}, \quad \alpha_i = \theta_i + a_{1,i}\ln a_{2,i} + \sum_{j\neq i} \theta_{i,j}\mathbb{E}_q[Z_j]$$

où il suffit de mettre à jour les $\mathbb{E}_q[Z_j]$ (numériquement pour les x_j non observés).

7.1.3 Inférence des hyperparamètres

On peut aussi utiliser une méthode variationnelle (cette fois l'algorithme EM classique) pour la phase de pré-traitement, c'est-à-dire pour l'inférence des hyper-paramètres $a_{0,i}$, $a_{1,i}$, $a_{2,i}$ et c_i. Pour le dernier qui est discret, on peut soit fixer une valeur, soit maximiser la vraisemblance pour plusieurs valeurs et garder la meilleure selon ce critère. Cela peut servir en particulier à estimer la puissance d'auto-activation $m_{ii} = a_{1,i}$ d'un gène à partir de sa loi marginale, comme cela a été fait pour WASABI. Si une puissance forte est trouvée, il peut s'agir soit d'une auto-activation directe, soit d'une boucle positive dont fait partie le gène. Par ailleurs, l'idée de cette phase est d'exploiter l'information temporelle lorsqu'elle est disponible, en autorisant seulement l'input extérieur θ à varier. Comme cette phase s'opère indépendamment pour chaque gène, on oublie temporairement l'indice i pour simplifier. Le modèle est le suivant, pour $c \in \mathbb{N}^*$ fixé (avec x observé et z caché) :

$$p(x,z|a,\theta_t) \propto \binom{c}{z} x^{-1}\exp\left(\theta_t z + a_0\ln x + a_1 z\ln x - a_2 x\right) \tag{7.4}$$

dont on rappelle la marginale

$$p(x) \propto x^{a_0-1}(1+e^{\theta_t}x^{a_1})^c e^{-a_2 x}$$

avec $\theta_t \in \mathbb{R}$ et $a_0, a_1, a_2 \in \mathbb{R}_+^*$. On suppose que θ_t peut varier dans le temps, contrairement aux autres paramètres. On a alors immédiatement :

$$\mathcal{L}(Z|X=x) = \mathcal{B}\left(c, \frac{e^{\theta_t}x^{a_1}}{1+e^{\theta_t}x^{a_1}}\right)$$

$$\mathcal{L}(X|Z=z) = \gamma\left(a_0 + a_1 z, a_2\right)$$

ce qui fournit par exemple un algorithme de Gibbs pour échantillonner selon $\mathcal{L}(X,Z)$. Un des paramètres les plus intéressants est $m = a_1$, qui correspond directement à la puissance d'auto-activation dans le modèle PDMP.

Algorithme EM. On pose, pour $\Theta^{(k)}$ fixé,

$$\Theta \mapsto Q(\Theta, \Theta^{(k)}) = \mathbb{E}_{\Theta^{(k)}}[\ln p(X,Z|\Theta)|X].$$

Grâce à la famille exponentielle, on obtient :

$$\begin{aligned}
\partial_{\theta_t} Q &= \mathbb{E}(Z|X,\Theta^{(k)}) - \mathbb{E}(Z|\Theta)\\
\partial_{a_0} Q &= \ln(X) - \mathbb{E}(\ln(X)|\Theta)\\
\partial_{a_1} Q &= \ln(X)\mathbb{E}(Z|X,\Theta^{(k)}) - \mathbb{E}(Z\ln(X)|\Theta)\\
\partial_{a_2} Q &= -X + \mathbb{E}(X|\Theta)
\end{aligned}$$

On remarque de plus que $\mathbb{E}(Z|X,\Theta) = c\frac{e^{\theta_t}X^{a_1}}{1+e^{\theta_t}X^{a_1}}$. Repartant de la loi jointe

$$p(x,z) = A^{-1}\binom{c}{z} e^{\theta_t z} x^{a_0+a_1 z-1} e^{-a_2 x}$$

avec

$$A = \sum_{z=0}^{c}\binom{c}{z} e^{\theta_t z}\frac{\Gamma(a_0+a_1 z)}{{a_2}^{a_0+a_1 z}} = \sum_{z=0}^{c}\binom{c}{z}\Gamma(a_0+a_1 z)e^{\theta_t z-(a_0+a_1 z)\ln(a_2)},$$

on a les formules suivantes, en posant $F(z) = A^{-1}\binom{c}{z}\Gamma(a_0+a_1 z)e^{\theta_t z-(a_0+a_1 z)\ln(a_2)}$:

- $\mathbb{E}(Z|\Theta) = \sum_{z=0}^{c} zF(z)$
- $\mathbb{E}(\ln X|\Theta) = \sum_{z=0}^{c}(\psi(a_0+a_1 z) - \ln(a_2))F(z)$
- $\mathbb{E}(Z\ln X|\Theta) = \sum_{z=0}^{c} z(\psi(a_0+a_1 z) - \ln(a_2))F(z)$
- $\mathbb{E}(X|\Theta) = \sum_{z=0}^{c}\frac{a_0+a_1 z}{a_2}F(z)$

L'algorithme EM correspond alors à l'itération des deux étapes suivantes :

1. trouver Θ_0 qui maximise $\Theta \mapsto Q(\Theta,\Theta^{(k)})$ pour $\Theta^{(k)}$ fixé
2. mettre à jour $\Theta^{(k)} = \Theta_0$

La Figure 7.1 montre un résultat typique de cette phase de pré-traitement.

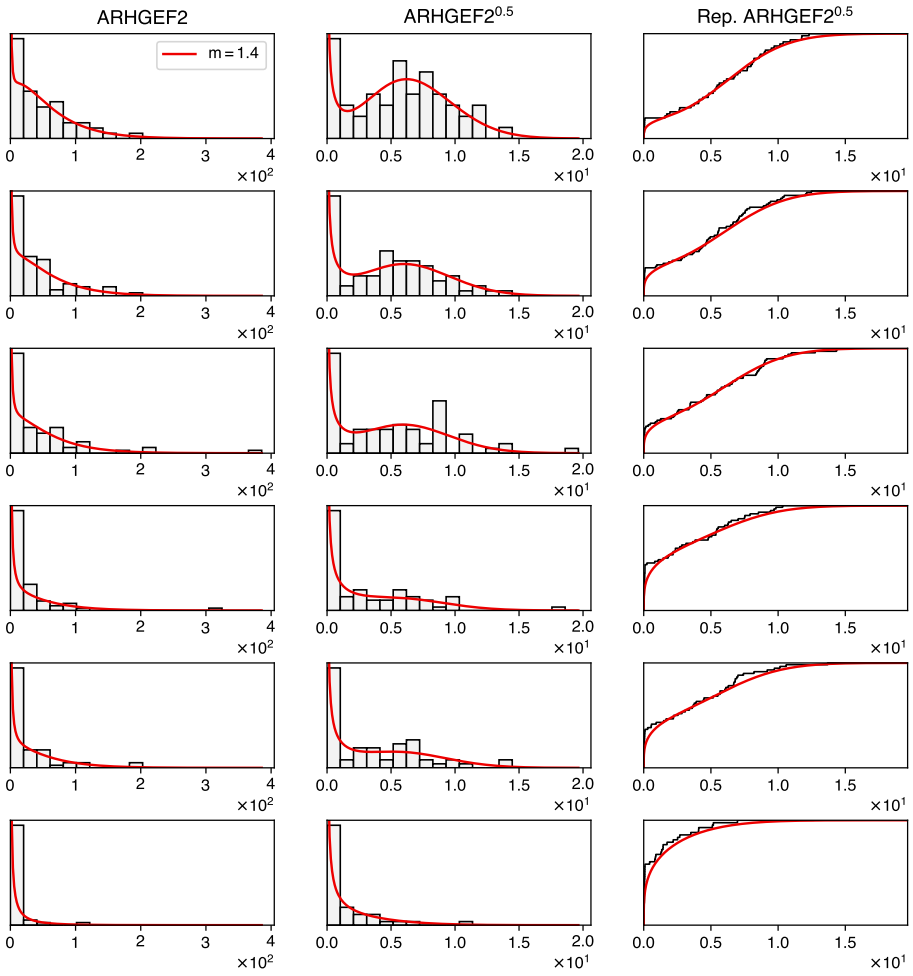


Figure 7.1 – Exemple de calibrage des hyperparamètres du modèle Gamma-Binomial pour un gène (ARHGEF2). Les courbes rouges représentent la distribution du modèle. Chaque ligne correspond à l'un des six time-points (0, 8, 24, 33, 48, 72 de haut en bas). Noter que ceux-ci sont utilisés simultanément pour l'inférence. La première colonne représente l'histogramme des données brutes, les deux autres sont basées sur les données transformées par l'application $x \mapsto \sqrt{x}$ (histogramme et fonction de répartition empirique). La puissance d'auto-activation trouvée ici est $m = 1.4$.

7.2 Premiers résultats sur des données simulées

Dans cette section, on présente quelques premiers résultats obtenus à partir de données simulées par le modèle de réseau PDMP, qui peut clairement faire office de « gold standard » pour les données *single-cell* en comparaison des méthodes de simulation plus classiques basées sur les données de population, qui sont très souvent des systèmes déterministes avec un bruit externe (Marbach *et al.*, 2010, 2012). Précisons aussi que pour inférer des graphes bien définis, nous avons ajouté à la fonction objectif Q précédente une pénalisation de type LASSO : il s'agit simplement de remplacer $Q(\theta, \alpha)$ par $Q(\theta, \alpha) - \lambda \sum_{i<j} |\theta_{ij}|$ où λ est le paramètre de pénalisation, de façon à ce que l'inférence privilégie les configurations où le plus de θ_{ij} possibles valent 0. Il est possible de garder un point de vue probabiliste en remarquant que cette pénalisation est équivalente à considérer l'estimateur du maximum a posteriori (MAP) pour une certaine distribution a priori sur les θ_{ij} (cf. 7.A.1 du chapitre précédent).

Fondamentalement, l'idée de l'auto-modèle Gamma-Binomial est la suivante : pour un instant t donné, il s'agit d'inférer un graphe non orienté qui résume les dépendances statistiques entre les gènes à cet instant. On adopte donc le point de vue d'un θ qui serait fonction de t, i.e. d'un graphe qui évolue dans le temps. Cela est assez différent de l'interprétation de départ qui se voulait mécaniste, mais néanmoins potentiellement utile car les graphes inférés peuvent aussi être interprétés comme un encodage « compressé » des lois jointes de n gènes par $n(n+1)/2$ paramètres (au lieu de $2^n - 1$ par exemple si l'on voulait décrire de manière générale la loi n de gènes à 2 niveaux chacun). On peut bien sûr faire le parallèle avec les vecteurs gaussiens qui ont exactement $n(n+1)/2$ paramètres (la matrice de covariance), mais il y a deux différences fondamentales : il s'agit d'un modèle caché, et la variable cachée est pressentie comme fondamentalement binaire plutôt que gaussienne.

D'autre part, nous montrons sur des exemples que l'on peut véritablement faire un lien fort entre l'observation d'une arrête $i - j$ à un certain instant et la présence effective d'une interaction $i \to j$ ou $j \to i$ dans le réseau de régulation (Figure 7.2). Nous avons opté pour une représentation graphique simple de la matrice symétrique θ inférée :

- chaque gène i possède une barre bleue dont la longueur correspond à la valeur de θ_i ;
- pour chaque paire de gènes $\{i, j\}$, on trace une arrête dont la couleur correspond au signe et l'épaisseur correspond à la valeur absolue de θ_{ij}.

Ainsi, l'absence d'arrête signifie que la valeur inférée de θ_{ij} est 0, et l'absence de barre bleue signifie que θ_i est très « petit » (typiquement négatif). Il convient de noter qu'un θ_i petit n'est pas synonyme d'un gène peu exprimé : un niveau d'expression élevé peut très bien être expliqué par des interactions positives fortes avec d'autres gènes, et c'est pour cela que θ_i ne doit pas être confondu avec le niveau d'expression moyen du gène.

À l'issue de cette inférence, il est possible d'analyser l'évolution de la structure du graphe pour tenter d'en déduire θ. Lorsque l'inférence se déroule bien et que l'on présume de surcroît une structure du réseau de type « voie de signalisation » (arbre orienté), il semble que cette phase de post-traitement puisse très bien se faire sur la base d'arguments logiques simples comme sur la Figure 7.2. Dans les cas où le réseau serait plus complexe (boucles de rétro-action) ou que les résultats de l'inférence soient bien moins clairs, il serait intéressant d'utiliser des approches existantes qui utilisent des arguments d'évolution temporelle en exploitant seulement l'évolution des lois jointes : c'est ce que WASABI fait dans une certaine mesure, mais c'est aussi la base de l'algorithme récemment développé baptisé SINCERITIES (Papili Gao *et al.*, 2018), qui est extrêmement rapide. L'intérêt serait alors de ne garder que les arrêtes inférées par l'auto-modèle Gamma-Binomial (en prenant par exemple l'union des graphes de chaque time-point), et ensuite de proposer une orientation de ces arrêtes – et seulement celles-ci – grâce à SINCERITIES.

7.3 Résultats sur les données réelles

Dans cette section, nous présentons des résultats très partiels d'inférence sur les données du chapitre 4. Avant toute considération biologique, un premier intérêt est de montrer que la méthode variationnelle fonctionne véritablement, et ce même pour un nombre relativement élevé de gènes (ici 90). À titre de comparaison, il a fallu environ une heure de calcul sur les serveurs de l'IN2P3 pour inférer les graphes associés aux six time-points, tandis que WASABI a nécessité deux semaines. Au cours des tests que nous avons pu faire sur les données simulées par le modèle PDMP, nous avons constaté une réelle robustesse de l'algorithme à partir d'un

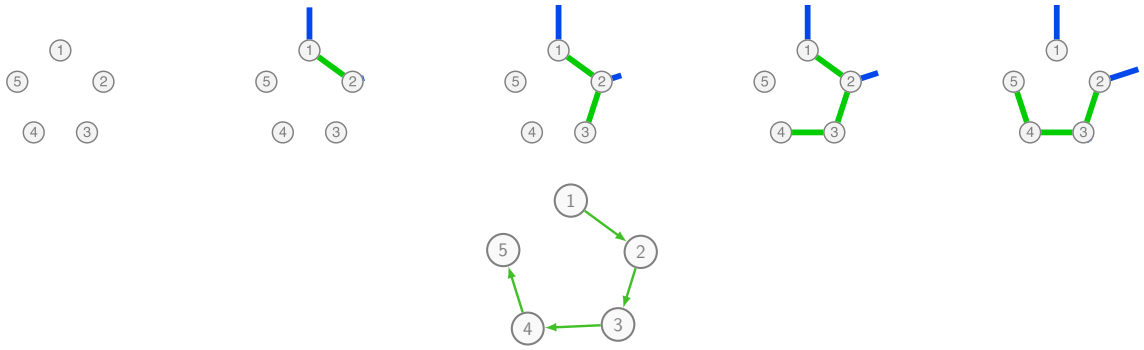


Figure 7.2 – Utilisation de l'auto-modèle Gamma-Binomial pour inférer un réseau de régulation à partir de données *snapshots* de cellules uniques. Le réseau (en bas) s'obtient ici simplement en observant l'ordre d'arrivée des dépendances statistiques entre les gènes.

nombre suffisant de données par rapport à la complexité du réseau simulé. Par exemple, un échantillon de 1000 cellules permet d'inférer de manière robuste un bon nombre de réseaux de 5 gènes qui ont une structure biologiquement vraisemblable. Mais en dessous de 100 cellules l'inférence est moins robuste : il convient dans ce cas d'augmenter le paramètre de pénalisation (ce qui à pour effet de diminuer le nombre d'arrêtes inférées mais d'offrir une meilleure garantie sur celles qui restent). Or le nombre de données réelles dont nous disposons est très faible en comparaison : après nettoyage des divers problèmes techniques, il ne reste qu'une soixantaine de cellules au maximum par time-point, ce qui nous a obligé à imposer une forte pénalisation.

Concernant les résultats à proprement parler, nous ne préciserons pas les noms des gènes pour éviter toute interprétation trop hâtive car cette approche est encore trop peu aboutie : nous nous focaliserons seulement sur un motif particulier qui semble émerger dans les données. La Figure 7.3 montre une représentation assez basique en cercle, l'avantage étant d'afficher systématiquement les 90 gènes observés en les laissant à la même place pour chaque time-point. La Figure 7.4 montre un autre type de représentation, plus facile au premier bord puisque qu'il ne montre que les gènes qui sont impliqués dans le graphe inféré. En revanche, il devient plus difficile de comparer les graphes au cours du temps.

On peut constater que la structure des arrêtes semble très instable dans le temps, ce qui nous fait pour le moment douter de la robustesse des graphes inférés. La première étape à effectuer après ce constat sera de faire du *bootstrap* sur les données pour voir si certaines arrêtes sont stables à un instant fixé lorsque les données sont légèrement perturbées – et dans ce cas ne retenir que celles-ci – ou bien si les arrêtes sont toutes instables, et dans ce cas il faudra manifestement augmenter le paramètre de pénalisation quitte à n'inférer que très peu d'arrêtes. À l'inverse, s'il s'avère que les arrêtes de chaque graphe sont robustes, cela signifierait que l'état biologique du réseau de gènes change bel et bien rapidement au cours de l'expérience. Remarquons que ce caractère fluctuant est déjà apparu en analysant les corrélations linéaires dans le chapitre 4, résistant effectivement au bootstrap. Dans tous les cas, ce type d'analyse pourrait se révéler assez riche d'enseignements, même si ce ne sont pas toujours ceux attendus.

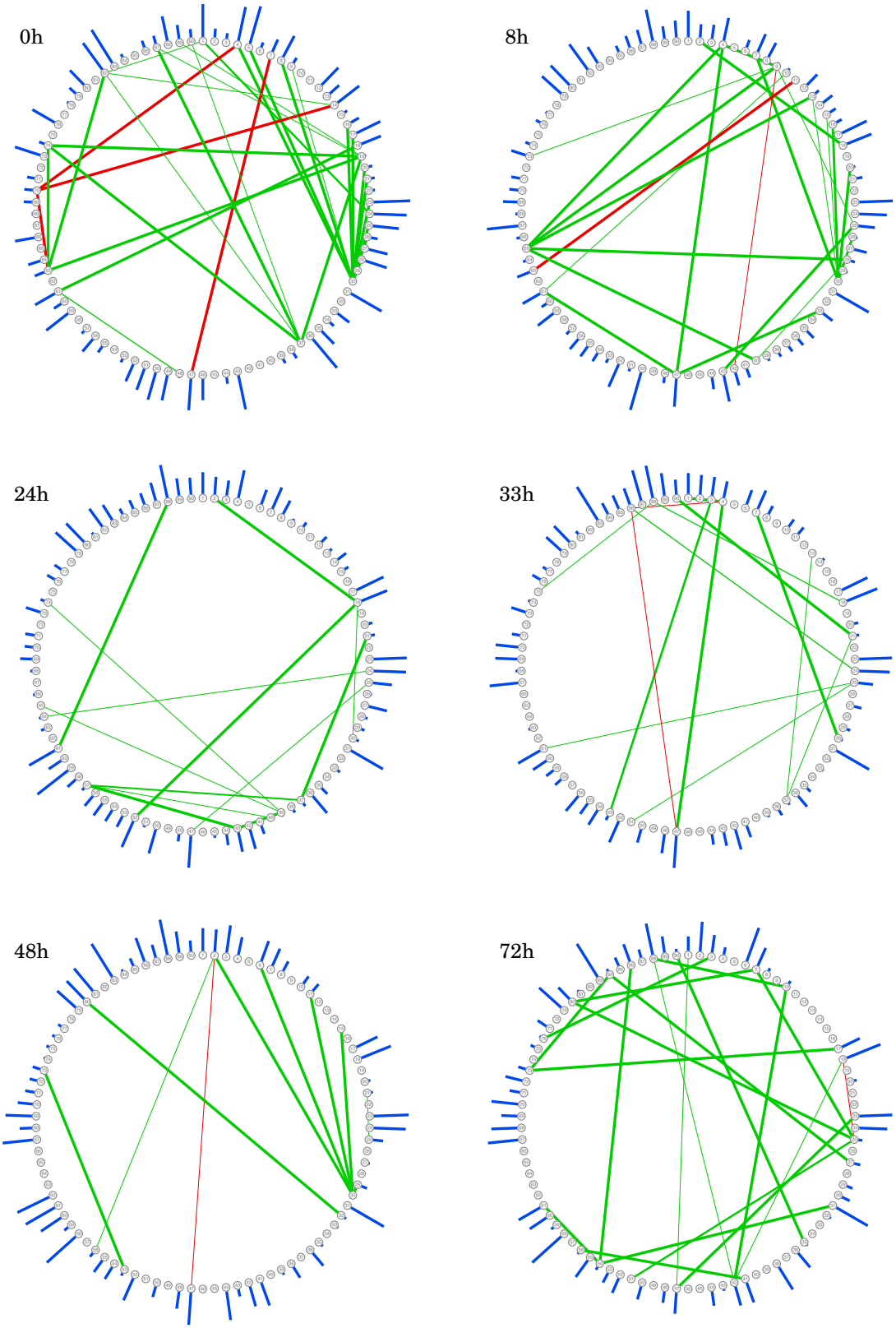


Figure 7.3 – Graphes inférés à partir des données du chapitre 4 aux six instants de mesure.

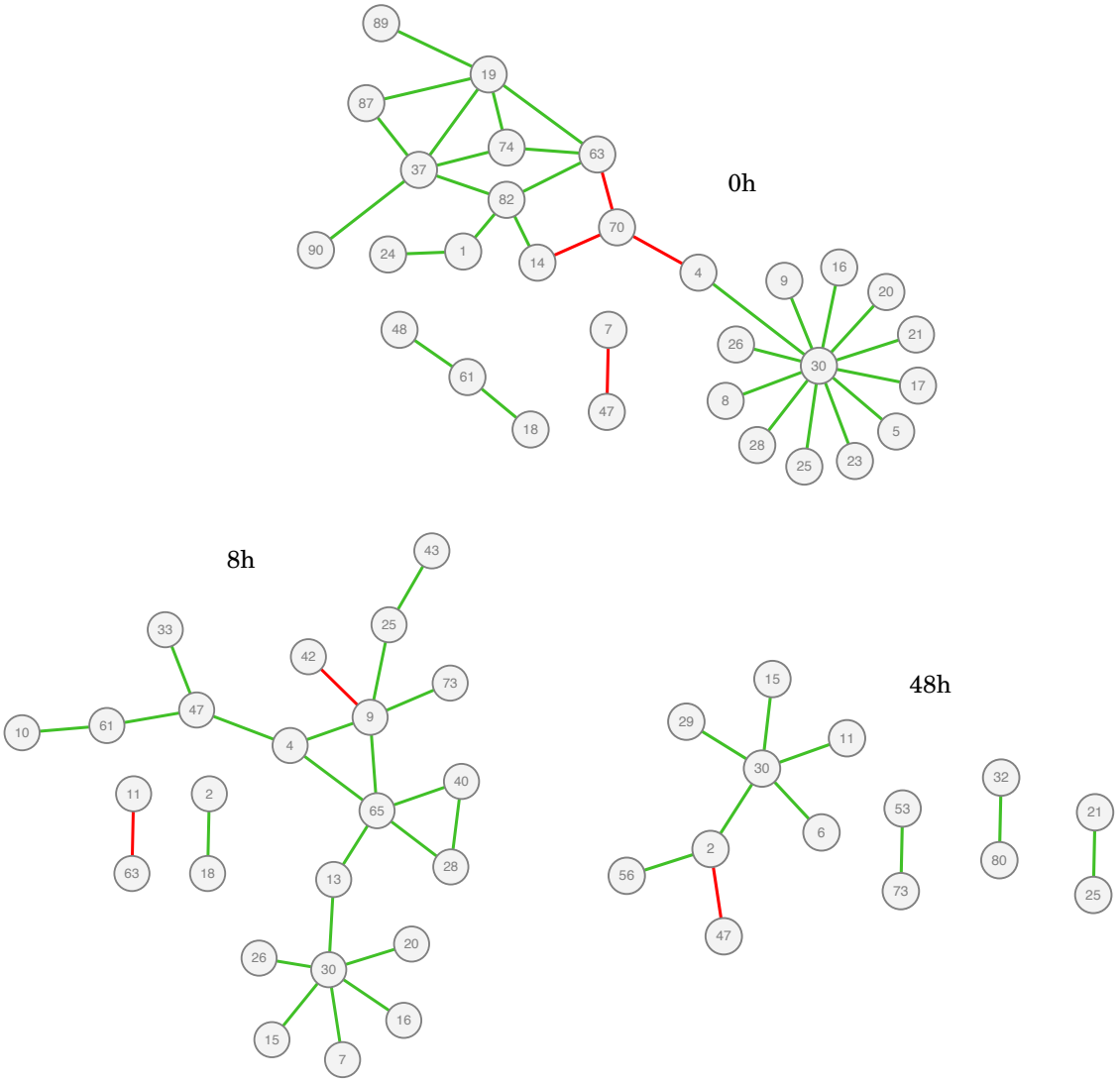


Figure 7.4 – Autre représentation pour les graphes inférés à partir des données du chapitre 4 aux instants $t = 0$, $t = 8$ et $t = 48$h. On constate que le gène 30 a souvent un motif d'interactions en étoile, et ce phénomène semble robuste dans le temps. À ce stade il est encore difficile de savoir s'il s'agit d'une réalité biologique ou d'un artefact technique, mais notre approche permet déjà de remarquer que cette structure est bel et bien présente dans les données. Noter que cela pourrait être en contradiction avec l'absence de *hubs* dans le réseau inféré par WASABI au chapitre 5.

Conclusion et perspectives

Mathematics is biology's next microscope, only better ;
biology is mathematics' next physics, only better. Joel E. Cohen (2004)

L'objectif général de cette thèse était de proposer un cadre pour l'inférence de réseaux de gènes en se basant sur des arguments biologiques plutôt qu'empiriques. Cela a débouché sur la construction d'un modèle dynamique prenant en compte le caractère fondamentalement stochastique de l'expression des gènes, ces derniers étant alors vus comme des particules en interaction probabiliste. Depuis qu'il est possible de mesurer les niveaux d'expression dans des cellules individuelles et non plus comme une moyenne sur un grand nombre de cellules, ce sujet est devenu central en biologie cellulaire (Symmons et Raj, 2016). En comparaison avec des approches plus traditionnelles où l'aléa est souvent considéré comme un bruit externe venant perturber une structure de dépendance déterministe, il est intéressant de constater que la source d'aléa biologique associée aux bursts de transcription permet à elle seule de décrire extrêmement bien la variabilité présente dans les observations. Il s'agit d'un fait bien connu – notamment à l'origine du succès rencontré par le modèle à deux états – mais encore très peu utilisé pour l'inférence, ce qui a motivé notre approche.

Par ailleurs, la communauté de biologie des systèmes qui s'intéresse au sujet est obligée de constater qu'après avoir inféré moult réseaux à partir de données de population, utilisant pour cela un arsenal statistique extrêmement large, elle ne parvient pas encore à incorporer de façon satisfaisante les données de cellules uniques (Chen et Mar, 2018). Cela peut sembler paradoxal puisque ces données sont par définition plus précises que les données de population : malgré les problèmes de variabilité technique qui leur sont souvent attribués, les données *single-cell* contiennent littéralement les données de population dans leur moyenne, ce qui est vérifiable quantitativement (Richard *et al.*, 2016). En fait, il est important de garder en tête que les réseaux inférés n'ont d'intérêt que s'ils permettent d'avancer dans la compréhension des phénomènes biologiques sous-jacents. Les méthodes d'inférence ont curieusement une forte tendance à l'oublier, en se focalisant régulièrement sur des performances théoriques à partir de données *in silico* qui n'ont parfois rien à voir avec les observations réelles. En outre, si les données de population ont pu rendre de grands services dans la détection de structures de régulation particulières, le fait que des données plus précises soient capables de mettre en défaut les modèles existants devrait inciter à une description plus fine, sachant que ce domaine de la biologie progresse à grands pas, notamment grâce à l'évolution spectaculaire des techniques d'observation (Battich *et al.*, 2013 ; Cremer *et al.*, 2015 ; Liu et Tjian, 2018). De façon plus concrète, on est passé d'un paradigme *linéaire gaussien* dans les données de population à un paradigme *non linéaire multimodal* dans les données de cellules uniques. Même si cela peut être frustrant au niveau mathématique de ne plus pouvoir utiliser le cadre gaussien, c'est aussi l'opportunité d'aller plus loin dans la modélisation et de s'intéresser à

des structures de dépendance statistique plus sophistiquées, vues par exemple comme les propriétés émergentes de modèles dynamiques.

Dans ce contexte, nous avons utilisé le formalisme des processus de Markov déterministes par morceaux qui semble constituer un excellent compromis entre simplicité mathématique et réalisme biologique, permettant de ne considérer que la source de bruit moléculaire la plus importante. Cela n'est pas spécifique aux réseaux de régulation : ces processus sont déjà populaires depuis un certain temps (Davis, 1984) et ils gagnent par exemple du terrain en physique statistique (Faggionato *et al.*, 2009) et en biologie de manière générale (Rudnicki et Tyran-Kamińska, 2017). Ils semblent en effet incarner assez bien le cheminement logique de la modélisation, lorsque la description d'un phénomène par des équations différentielles s'avère insuffisante et qu'il est nécessaire de prendre en compte un aspect aléatoire allant au delà du simple « bruit blanc » associé au mouvement brownien. Remarquons aussi que l'aspect multi-échelle apparaît inévitablement au sujet des réseaux de régulation (espèces rares/abondantes, dynamiques lentes/rapides, etc.). En particulier, il est possible d'identifier trois échelles distinctes où l'aspect binaire joue manifestement un rôle important : l'échelle de la configuration permissive/non permissive de la chromatine, celle de l'état actif/inactif du promoteur, et enfin l'échelle de la fréquence d'allumage haute/basse de ce même promoteur qui est apparue en modélisant les interactions (cf. chapitre 6).

Concernant nos résultats, les trois méthodes d'inférence présentées dans cette thèse sont intéressantes avec des avantages et des inconvénients pour chacune :

- L'algorithme WASABI est très flexible et permet de combiner des données hétérogènes (demi-vies, niveaux d'ARNm et de protéines, cellules uniques et population) ainsi que l'information temporelle des données. Elle nécessite par contre une grosse capacité de calcul en raison de son caractère *brute-force*, et surtout elle n'utilise à l'heure actuelle que les distributions marginales des niveaux d'expression ;
- L'approximation de Hartree (avec l'algorithme *hard EM* pour l'inférence) est la méthode la plus rapide et peut potentiellement se baser sur n'importe quelle forme d'interaction sous-jacente, mais en faisant l'hypothèse forte que les données suivent la loi stationnaire du modèle PDMP. En outre, sa forme de pseudo-vraisemblance fait que l'identifiabilité n'est pas garantie, et il n'est pas clair qu'elle puisse véritablement retrouver le sens des interactions sans apport d'information supplémentaire.
- L'auto-modèle Gamma-Binomial (avec la méthode d'inférence variationnelle) est assez rapide également et constitue une forme d'approximation ultime du PDMP, donnant à cette approche un caractère plus descriptif (i.e. phénoménologique) que les précédentes. En contrepartie, on a la garantie que le modèle est identifiable tout en étant capable de reproduire assez bien les données. Il permet seulement d'inférer des graphes non orientés résumant les dépendances statistiques entre les gènes à chaque instant, ce qui est plus cohérent que le point de vue stationnaire de départ, mais un post-traitement est nécessaire si l'on veut en déduire un réseau de régulation.

Au vu de l'équivalence entre l'approximation de Hartree et la pseudo-vraisemblance de Besag dans le cas résoluble (Corollaire 2.17), il est assez probable que la deuxième méthode soit en fait une approximation numérique plus ou moins précise de la troisième. De notre point de vue, il serait extrêmement intéressant de pouvoir combiner l'auto-modèle Gamma-Binomial avec l'algorithme WASABI, de façon à ce que ce dernier n'exploite pas seulement les marginales mais aussi les lois jointes. Une autre option prometteuse serait d'utiliser la version temporelle de l'approximation de Hartree, ce qui nécessiterait une étape d'approximation

supplémentaire mais est tout à fait envisageable. De manière générale, nous plaidons pour ce type d'approches fondées sur une description biologique comme des concurrents sérieux de méthodes non paramétriques issues de la théorie de l'information qui sont développées en parallèle (Chan *et al.*, 2017), et ce pour deux raisons principales :

- Dans un contexte ou le nombre de données est souvent bien plus petit que le nombre de paramètres, les modèles « paramétriques » basés sur des connaissances biologiques ont de grandes chances d'être plus robustes que les modèles « non paramétriques » qui ont en fait un nombre bien plus grand de degrés de liberté. À titre d'exemple, pour $n = 100$ gènes avec seulement 2 niveaux possibles, une méthode non paramétrique « agnostique » devrait commencer par estimer les $2^n - 1 \approx 10^{30}$ paramètres caractérisant les distributions sur $\{0, 1\}^n$, tandis que l'auto-modèle Gamma-Binomial est entièrement caractérisé par $n(n + 1)/2 + 3n = 5350$ paramètres et hyperparamètres.
- Les méthodes en circulation n'exploitent jamais la loi jointe complète – notamment à cause du premier point – mais seulement les lois de couples ou de triplets de gènes. L'ajout d'interactions se fait alors de manière indépendante pour chaque arrête sur la base d'un score, ce qui crée un risque de redondance (cf. section 7.2) tout en soulevant des problèmes bien connus liés aux procédures de tests multiples.

De plus, il apparaît clairement qu'il faut assez de données par rapport à la complexité présumée des réseaux que l'on souhaite inférer : celles que nous avons utilisées ici ont été obtenues par RT-qPCR, une technique qui fournit peu de cellules (moins d'une centaine par *time-point*) mais avec des mesures précises. À l'opposé, les données de type RNA-seq sont moins précises mais beaucoup plus abondantes, atteignant récemment le million de cellules avec la technologie SPLiT-Seq (Svensson *et al.*, 2018). Il serait très intéressant d'appliquer nos différentes stratégies sur ces données.

La théorie quantique des champs appliquée à la biologie

C'est une alliance improbable mais prometteuse. Comme nous l'avons mentionné au chapitre 2, ce sont les physiciens Sasai et Wolynes (2003) qui ont introduit l'idée que les réseaux de gènes pouvaient être modélisés mathématiquement en décrivant chaque gène de la même façon qu'une particule quantique, et plus spécifiquement grâce à un modèle spin-boson. Même s'il n'y a rien de physique dans cette analogie puisque les échelles considérées sont complètement différentes, ce point de vue a permis d'adapter une méthode classique de théorie quantique des champs à l'étude des réseaux de gènes, débouchant sur l'approximation *Self-Consistent Proteomic Field* (Walczak *et al.*, 2005) que nous avons adaptée dans cette thèse à notre formalisme PDMP sous le nom d'approximation de Hartree. Il y a également un changement essentiel d'un point de vue conceptuel : un état du réseau n'est plus un *point* comme ce serait le cas avec un système d'équations différentielles, mais une *distribution* comme c'est le cas pour les particules quantiques, représentant ici les probabilités des niveaux d'expression à un instant donné (Figure 3). Pour faire l'analogie jusqu'au bout, deux gènes qui présentent des dépendances au niveau probabiliste dans leurs niveaux d'expression se comportent donc comme des particules quantiques dont les états sont intriqués.

D'un point de vue mathématique, cette analogie est également intéressante car elle suggère de réfléchir en termes d'analyse fonctionnelle, notamment via le semi-groupe d'opérateurs associé à l'équation maîtresse. On a exploité ce formalisme dans le chapitre 3, ce qui a permis d'interpréter la représentation de Poisson en termes de sous-espace vectoriel stable

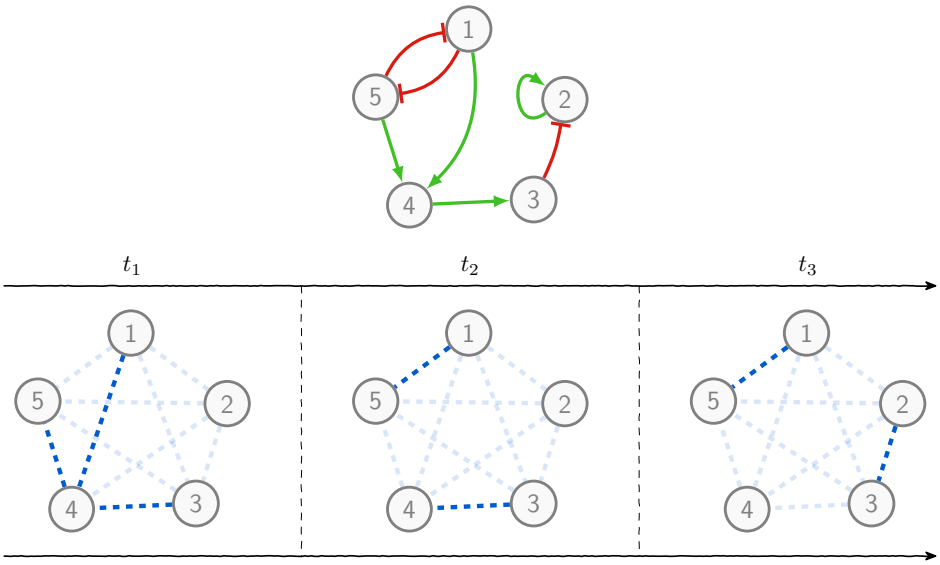


Figure 3 – *Structure* du réseau (en haut) vs. *états* du réseau (en bas). Les états sont des distributions dont on a symbolisé ici la structure de dépendance à la manière des champs de Markov.

pour le semi-groupe. Une différence fondamentale avec la physique quantique est que l'opérateur « hamiltonien » (le générateur du semigroupe de l'équation maîtresse, i.e. l'adjoint du générateur du processus à proprement parler) n'est typiquement pas symétrique, ce qui complique évidemment une éventuelle approche spectrale. Il serait à notre sens intéressant d'explorer plus en détail ces aspects, en utilisant par exemple des outils d'origine probabiliste comme le couplage de PDMP (Benaïm *et al.*, 2012).

Une émancipation progressive des cellules

Finalement, on remarque un phénomène particulier qui semble lent mais inexorable : les cellules sont de plus en plus perçues comme des entités autonomes, dont le comportement n'est pas entièrement prévisible. En particulier, une quantité croissante d'indices suggère que la variabilité de l'expression des gènes et la possibilité de transmission de motifs d'expression aux cours des générations de cellules sont tout à fait capables de produire une forme de sélection naturelle au sein d'un organisme, sans être basée sur des mutations génétiques (Heams, 2004 ; Huang, 2010). Cela nous permet de revenir sur la pseudo-citation de Stephen Hawking en introduction : on peut envisager que la différenciation soit une forme d'*adaptation* des cellules à leur milieu. La variabilité pourrait alors avoir un rôle fonctionnel en facilitant l'exploration d'un paysage épigénétique, hypothèse désormais sérieuse et qui a été notamment abordée au chapitre 4, mais qui devrait nécessiter encore de nombreuses avancées techniques et conceptuelles avant de pouvoir être confirmée.

Bibliographie

Albayrak, C., Jordi, C. A., Zechner, C., Lin, J., Bichsel, C. A., Khammash, M., et Tay, S. Digital Quantification of Proteins and mRNA in Single Mammalian Cells. *Molecular Cell*, 61:914–924, 2016.

Anderson, D. F. et Kurtz, T. G. *Stochastic Analysis of Biochemical Systems*, volume 1.2 de *Mathematical Biosciences Institute Lecture Series*. Springer International Publishing, 2015.

Babtie, A. C., Chan, T. E., et Stumpf, M. P. Learning regulatory models for cell development from single cell transcriptomic data. *Current Opinion in Systems Biology*, 5:72–81, 2017.

Barabási, A.-L. et Oltvai, Z. N. Network biology: understanding the cell's functional organization. *Nature Reviews Genetics*, 5:101–113, 2004.

Bartlett, T. E., Müller, S., et Diaz, A. Single-cell coexpression subnetwork analysis. *Sci Rep*, 7(1), 2017.

Battich, N., Stoeger, T., et Pelkmans, L. Image-based transcriptomics in thousands of single human cells at single-molecule resolution. *Nature Methods*, 10:1127–1133, 2013.

Becskei, A., Kaufmann, B. B., et van Oudenaarden, A. Contributions of low molecule number and chromosomal positioning to stochastic gene expression. *Nature Genetics*, 37(9), 2005.

Benaïm, M., Le Borgne, S., Malrieu, F., et Zitt, P.-A. Quantitative ergodicity for some switched dynamical systems. *Electronic Communications in Probability*, 17(56):1–14, 2012.

Benaïm, M., Le Borgne, S., Malrieu, F., et Zitt, P.-A. Qualitative properties of certain piecewise deterministic Markov processes. *Annales de l'Institut Henri Poincaré - Probabilités et Statistiques*, 51(3):1040–1075, 2015.

Besag, J. Spatial Interaction and the Statistical Analysis of Lattice Systems. *Journal of the Royal Statistical Society*, 36(2):192–236, 1974.

Besag, J. Statistical Analysis of Non-Lattice Data. *The Statistician*, 24(3):179–195, 1975.

Bintu, L., Yong, J., Antebi, Y. E., McCue, K., Kazuki, Y., Uno, N., Oshimura, M., et Elowitz, M. B. Dynamics of epigenetic regulation at the single-cell level. *Science*, 351(6274), 2016.

Blei, D. M., Ng, A. Y., et Jordan, M. I. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

Boettiger, A. N. Analytic Approaches to Stochastic Gene Expression in Multicellular Systems. *Biophysical Journal*, 105(12):2629 – 2640, 2013.

Boxma, O., Kaspi, H., Kella, O., et Perry, D. On/Off Storage Systems with State-Dependent Input, Output, and Switching Rates. *Probability in the Engineering and Informational Sciences*, 19:1–14, 2005.

Celeux, G. et Govaert, G. A classification EM algorithm for clustering and two stochastic versions. Research Report 1364, INRIA, 1991.

Chalancon, G., Ravarani, C., Balaji, S., Martinez-Arias, A., Aravind, L., Jothi, R., et Babu, M. M. Interplay between gene expression noise and regulatory network architecture. *Trends Genet.*, 28(5):221–232, 2012.

Chan, T. E., Stumpf, M. P. H., et Babtie, A. C. Gene regulatory network inference from single-cell data using multivariate information measures. *Cell Systems*, 5(3), 2017.

Chen, S. et Mar, J. C. Evaluating methods of inferring gene regulatory networks highlights their lack of performance for single cell gene expression data. *BMC Bioinformatics*, 19(232), 2018.

Chong, S., Chen, C., Ge, H., et Xie, X. S. Mechanism of transcriptional bursting in bacteria. *Cell*, 158(2), 2014.

Cohen, J. E. Mathematics is biology's next microscope, only better; biology is mathematics' next physics, only better. *PLOS Biology*, 2(12), 2004.

Coulon, A., Chow, C. C., Singer, R. H., et Larson, D. R. Eukaryotic transcriptional dynamics: from single molecules to cell populations. *Nature Reviews Genetics*, 14(8), 2013.

Coulon, A., Gandrillon, O., et Beslon, G. On the spontaneous stochastic dynamics of a single gene: complexity of the molecular interplay at the promoter. *BMC Systems Biology*, 4(2), 2010.

Cremer, T., Cremer, M., Hübner, B., Strickfaden, H., Smeets, D., Popken, J., Sterr, M., Markaki, Y., Rippe, K., et Cremer, C. The 4d nucleome: Evidence for a dynamic nuclear landscape based on co-aligned active and inactive nuclear compartments. *FEBS Lett*, 589:2931–2943, 2015.

Crudu, A., Debussche, A., Muller, A., et Radulescu, O. Convergence of stochastic gene networks to hybrid piecewise deterministic processes. *The Annals of Applied Probability*, 22(5):1822–1859, 2012.

Dattani, J. *Exact solutions of master equations for the analysis of gene transcription models*. Thèse de doctorat, Imperial College London, 2016.

Dattani, J. et Barahona, M. Stochastic models of gene transcription with upstream drives: exact solution and sample path characterization. *Journal of the Royal Society, Interface*, 14(126), 2017.

Davila-Velderrain, J., Martínez-García, J. C., et Alvarez-Buylla, E. R. Modeling the epigenetic attractors landscape: Towards a post-genomic mechanistic understanding of development. *Frontiers in Genetics*, 6(160), 2015.

Davis, M. H. A. Piecewise-deterministic markov processes: A general class of non-diffusion stochastic models. *Journal of the Royal Statistical Society*, 46(3):353–388, 1984.

Dufresne, D. The beta product distribution with complex parameters. *Communications in Statistics—Theory and Methods*, 39(5):837–854, 2010.

Dunkl, C. F. Products of Beta distributed random variables. *arXiv preprint*, 2013.

Eldar, A. et Elowitz, M. B. Functional roles for noise in genetic circuits. *Nature*, 467(7312):167–173, 2010.

Faggionato, A., Gabrielli, D., et Crivellari, M. Averaging and large deviation principles for fully-coupled piecewise deterministic markov processes and applications to molecular motors. *Markov Processes and Related Fields*, 16(3):497–548, 2010.

Faggionato, A., Gabrielli, D., et Crivellari, M. R. Non-equilibrium thermodynamics of piecewise deterministic markov processes. *Journal of Statistical Physics*, 137:259–304, 2009.

Feller, W. On a General Class of "Contagious" Distributions. *The Annals of Mathematical Statistics*, 14(4):389–400, 1943.

Fourel, G., Magdinier, F., et Gilson, E. Insulator dynamics and the setting of chromatin domains. *BioEssays*, 26(5), 2004.

Friedman, N., Cai, L., et Xie, X. S. Linking stochastic dynamics to population distribution: an analytical framework of gene expression. *Phys Rev Lett*, 97(16), 2006.

Fukaya, T., Lim, B., et Levine, M. Enhancer control of transcriptional bursting. *Cell*, 166(2), 2016.

Gallopin, M., Rau, A., et Jaffrézic, F. A hierarchical poisson log-normal model for network inference from rna sequencing data. *PLOS ONE*, 8(10):e77503, 2013.

Gardiner, C. W. et Chaturvedi, S. The Poisson Representation. I. A New Technique for Chemical Master Equations. *Journal of Statistical Physics*, 17(6):429–468, 1977.

Ghazanfar, S., Bisogni, A. J., Ormerod, J. T., Lin, D. M., et Yang, J. Y. H. Integrated single cell data analysis reveals cell specific networks and novel coactivation markers. *BMC Systems Biology*, 10(Suppl 5), 2016.

Gillespie, D. T. Exact stochastic simulation of coupled chemical reactions. *The Journal of Physical Chemistry*, 81(25):2340–2361, 1977.

Gu, J., Gu, Q., Wang, X., Yu, P., et Lin, W. Sphinx: modeling transcriptional heterogeneity in single-cell RNA-Seq. *bioRxiv preprint*, 2015.

Haddad, N., Jost, D., et Vaillant, C. Perspectives: using polymer modeling to understand the formation and function of nuclear compartments. *Chromosome Res*, 25(1):1–16, 2017.

Halpern, K. B., Tanami, S., Landen, S., Chapal, M., Szlak, L., Hutzler, A., Nizhberg, A., et Itzkovitz, S. Bursty gene expression in the intact mammalian liver. *Molecular Cell*, 58:147–156, 2015.

Hathaway, N. A., Bell, O., Hodges, C., Miller, E. L., Neel, D. S., et Crabtree, G. R. Dynamics and memory of heterochromatin in living cells. *Cell*, 149(7), 2012.

Heams, T. *An endodarwinian approach of intercellular variability of gene expression*. Thèse de doctorat, INAPG (AgroParisTech), 2004.

Hecker, M., Lambeck, S., Toepfer, S., van Someren, E., et Guthke, R. Gene regulatory network inference: data integration in dynamic models-a review. *BioSystems*, 96(1), 2009.

Herbach, U., Bonnaffoux, A., Espinasse, T., et Gandrillon, O. Inferring gene regulatory networks from single-cell data: a mechanistic approach. *BMC Syst Biol*, 11(1), 2017.

Hille, S. C. et Worm, D. T. Embedding of Semigroups of Lipschitz Maps into Positive Linear Semigroups on Ordered Banach Spaces Generated by Measures. *Integral Equations and Operator Theory*, 63:351–371, 2009.

Huang, S. Non-genetic heterogeneity of cells in development: more than just noise. *Development*, 136(23), 2009.

Huang, S. Cell lineage determination in state space: a systems view brings flexibility to dogmatic canonical rules. *PLOS Biology*, 8(5), 2010.

Huynh-Thu, V. A., Irrthum, A., Wehenkel, L., et Geurts, P. Inferring regulatory networks from expression data using tree-based methods. *PLOS One*, 5(9), 2010.

Innocentini, G. d. C. P., Forger, M., Ramos, A. F., Radulescu, O., et Hornos, J. E. M. Multimodality and flexibility of stochastic gene expression. *Bull Math Biol*, 75(12), 2013.

Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., et Saul, L. K. An Introduction to Variational Methods for Graphical Models. *Machine Learning*, 37(2):183–233, 1999.

Kaern, M., Elston, T. C., Blake, W. J., et Collins, J. J. Stochasticity in gene expression: from theories to phenotypes. *Nat Rev Genet*, 6(6):451–464, 2005.

Kim, J. K. et Marioni, J. C. Inferring the kinetics of stochastic gene expression from single-cell RNA-sequencing data. *Genome Biology*, 14:R7, 2013.

Kim, K.-Y. et Wang, J. Potential energy landscape and robustness of a gene regulatory network: toggle switch. *PLOS Computational Biology*, 3(3), 2007.

Klenke, A. *Probability Theory: a comprehensive course*. Springer, 2014.

Ko, M. S. H. A stochastic model for gene induction. *Journal of Theoretical Biology*, 153:181–194, 1991.

Ko, M. S. H., Nakauchi, H., et Takahashi, N. The dose dependence of glucocorticoid-inducible gene expression results from changes in the number of transcriptionally active templates. *The EMBO Journal*, 9(9):2835–2842, 1990.

Kueng, S., Oppikofer, M., et Gasser, S. M. Sir proteins and the assembly of silent chromatin in budding yeast. *Annual Review of Genetics*, 47, 2013.

Laforge, B., Guez, D., Martinez, M., et Kupiec, J.-J. Modeling embryogenesis and cancer: an approach based on an equilibrium between the autostabilization of stochastic gene expression and the interdependence of cells for proliferation. *Prog Biophys Mol Biol*, 89(1):93–120, 2005.

Larson, D. R. What do expression dynamics tell us about the mechanism of transcription? *Curr Opin Genet Dev*, 21(5):591–599, 2011.

Li, C. et Wang, J. Quantifying waddington landscapes and paths of non-adiabatic cell fate decisions for differentiation, reprogramming and transdifferentiation. *J R Soc Interface*, 10(89), 2013.

Liggett, T. M. *Interacting Particle Systems*. Springer-Verlag Berlin Heidelberg, 2005.

Liggett, T. M. *Continuous Time Markov Processes: An Introduction*. American Mathematical Society, 2010.

Liu, Z. et Tjian, R. Visualizing transcription factor dynamics in living cells. *J Cell Biol*, 217(4):1–11, 2018.

Mackey, M. C., Tyran-Kamińska, M., et Yvinec, R. Molecular distributions in gene regulatory dynamics. *Journal of Theoretical Biology*, 274(1), 2011.

Malrieu, F. Some simple but challenging Markov processes. *Annales de la Faculté de Sciences de Toulouse*, 24(4):857–883, 2015.

Marbach, D., Costello, J. C., Küffner, R., Vega, N. M., Prill, R. J., Camacho, D. M., Allison, K. R., DREAM5 Consortium, Kellis, M., Collins, J. J., et Stolovitzky, G. Wisdom of crowds for robust gene network inference. *Nature Methods*, 9(8), 2012.

Marbach, D., Prill, R. J., Schaffter, T., Mattiussi, C., Floreano, D., et Stolovitzky, G. Revealing strengths and weaknesses of methods for gene network inference. *PNAS*, 107(14), 2010.

Mazza, C. et Benaïm, M. *Stochastic Dynamics for Systems Biology*. CRC Press, 2014.

Mirny, L. A. Nucleosome-mediated cooperativity between transcription factors. *PNAS*, 107(52), 2010.

Mizeranschi, A., Zheng, H., Thompson, P., et Dubitzky, W. Evaluating a common semi-mechanistic mathematical model of gene-regulatory networks. *BMC Systems Biology*, 9(5):1–12, 2015.

Moris, N., Pina, C., et Arias, A. M. Transition states and cell fate decisions in epigenetic landscapes. *Nature Reviews Genetics*, 17(11), 2016.

Murphy, K. P. *Machine learning: a probabilistic perspective*. The MIT Press, 2012.

Pájaro, M., Alonso, A. A., Otero-Muras, I., et Vázquez, C. Stochastic modeling and numerical simulation of gene regulatory networks with protein bursting. *Journal of Theoretical Biology*, 421, 2017.

Pakdaman, K., Thieullen, M., et Wainrib, G. Asymptotic expansion and central limit theorem for multiscale piecewise-deterministic Markov processes. *Stochastic Processes and their Applications*, 122:2292–2318, 2012.

Papili Gao, N., Ud-Dean, S. M. M., Gandrillon, O., et Gunawan, R. SINCERITIES: Inferring gene regulatory networks from time-stamped single cell transcriptional expression profiles. *Bioinformatics*, 34(2):258–266, 2018.

Parikh, N. et Boyd, S. Proximal algorithms. *Foundations and Trends in Optimization*, 1(3):123–231, 2013.

Pazy, A. *Semigroups of Linear Operators and Applications to Partial Differential Equations*. Springer, 1983.

Peccoud, J. et Ycart, B. Markovian Modelling of Gene Product Synthesis. *Theoretical Population Biology*, 48:222–234, 1995.

Raj, A., Peskin, C. S., Tranchina, D., Vargas, D. Y., et Tyagi, S. Stochastic mRNA Synthesis in Mammalian Cells. *PLOS Biology*, 4(10), 2006.

Raj, A. et van Oudenaarden, A. Nature, nurture, or chance: stochastic gene expression and its consequences. *Cell*, 135(2), 2008.

Rao, S. S. P., Huntley, M. H., Durand, N. C., Stamenova, E. K., Bochkov, I. D., Robinson, J. T., Sanborn, A. L., Machol, I., Omer, A. D., Lander, E. S., et Aiden, E. L. A 3d map of the human genome at kilobase resolution reveals principles of chromatin looping. *Cell*, 159(7):1665–1680, 2014.

Raser, J. M. et O'Shea, E. K. Control of stochasticity in eukaryotic gene expression. *Science*, 304(5678), 2004.

Richard, A., Boullu, L., Herbach, U., Bonnafoux, A., Morin, V., Vallin, E., Guillemin, A., Papili Gao, N., Gunawan, R., Cosette, J., Arnaud, O., Kupiec, J.-J., Espinasse, T., Gonin-Giraud, S., et Gandrillon, O. Single-cell-based analysis highlights a surge in cell-to-cell molecular variability preceding irreversible commitment in a differentiation process. *PLOS Biology*, 14(12), 2016.

Rudnicki, R. On a stochastic gene expression with pre-mRNA, mRNA and protein contribution. *Journal of Theoretical Biology*, 2015.

Rudnicki, R. et Tyran-Kamińska, M. *Piecewise Deterministic Processes in Biological Models*. SpringerBriefs in Applied Sciences and Technology - Mathematical Methods. Springer Nature, 2017.

Sasai, M. et Wolynes, P. G. Stochastic gene expression as a many-body problem. *PNAS*, 100(5):2374–2379, 2003.

Schwanhäusser, B., Busse, D., Li, N., Dittmar, G., Schuchhardt, J., Wolf, J., Chen, W., et Selbach, M. Global quantification of mammalian gene expression control. *Nature*, 495, 2011.

Senecal, A., Munsky, B., Proux, F., Ly, N., Braye, F. E., Zimmer, C., Mueller, F., et Darzacq, X. Transcription Factors Modulate c-Fos Transcriptional Bursts. *Cell Reports*, 8:75–83, 2014.

Singer, Z. S., Yong, J., Tischler, J., Hackett, J. A., Altinok, A., Surani, M. A., Cai, L., et Elowitz, M. B. Dynamic heterogeneity and dna methylation in embryonic stem cells. *Molecular Cell*, 55(2), 2014.

Slater, L. J. *Generalized hypergeometric functions*. Cambridge University Press, 1966.

Springer, M. D. et Thompson, W. E. The distribution of products of beta, gamma and gaussian random variables. *SIAM Journal on Applied Mathematics*, 18(4):721–737, 1970.

Suter, D. M., Molina, N., Gatfield, D., Schneider, K., Schibler, U., et Naef, F. Mammalian genes are transcribed with widely different bursting kinetics. *Science*, 332(6028), 2011.

Svensson, V., Vento-Tormo, R., et Teichmann, S. A. Exponential scaling of single-cell RNA-seq in the past decade. *Nature Protocols*, 13:599–604, 2018.

Symmons, O. et Raj, A. What's luck got to do with it: Single cells, multiple fates, and biological nondeterminism. *Mol Cell*, 62(5), 2016.

Viñuelas, J., Kaneko, G., Coulon, A., Vallin, E., Morin, V., Mejia-Pous, C., Kupiec, J.-J., Beslon, G., et Gandrillon, O. Quantifying the contribution of chromatin dynamics to stochastic gene expression reveals long, locus-dependent periods between transcriptional bursts. *BMC Biology*, 11(1), 2013.

Wainwright, M. J. et Jordan, M. I. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1(1-2):1–305, 2008.

Walczak, A. M., Sasai, M., et Wolynes, P. G. Self-consistent proteomic field theory of stochastic gene switches. *Biophysical Journal*, 88:828–850, 2005.

Wang, Y. R. et Huang, H. Review on statistical methods for gene network reconstruction using expression data. *Journal of Theoretical Biology*, 362:53–61, 2014.

Zhang, B. et Wolynes, P. G. Stem cell differentiation as a many-body problem. *PNAS*, 111(28):10185–10190, 2014.

Zhou, J. X., Aliyu, M. D. S., Aurell, E., et Huang, S. Quasi-potential landscape in complex multi-stable systems. *Journal of The Royal Society Interface*, 2012.

Zoller, B., Nicolas, D., Molina, N., et Naef, F. Structure of silent transcription intervals and noise characteristics of mammalian genes. *Mol Syst Biol*, 11(7), 2015.