Inférence de réseaux de régulation de gènes à partir de données dynamiques multi-échelles

Arnaud Bonnaffoux

Numéro National de Thèse : 2018LYSEN054

THESE de DOCTORAT DE L'UNIVERSITE DE LYON
opérée par
l'Ecole Normale Supérieure de Lyon

Ecole Doctorale N°340
Biologie Moléculaire Intégrative et Cellulaire

Discipline : Biologie

Soutenue publiquement le 12/10/2018, par :

Arnaud BONNAFFOUX

Inférence de réseaux de régulation de gènes à partir de données dynamiques multi-échelles

Devant le jury composé de :

DUPONT Geneviève	Professeure des universités, ULB	Rapporteure
SIEGEL Anne	Directrice de recherche, IRISA	Rapporteure
BATT Gregory	Directeur de recherche, Institut Pasteur	Examinateur
GROS Pierre-Alexis	Senior Lead Scientist, Cosmo Tech	Examinateur
GANDRILLON Olivier	Directeur de recherche, LBMC ENS de Lyon	Directeur de thèse

Remerciements

Ces 3 années de thèse ont été pour moi une formidable aventure scientifique et humaine. Je tiens à remercier toutes les personnes qui ont contribué à ces travaux ou qui m'ont accompagné dans ce projet.

Je commencerai par remercier Olivier Gandrillon, mon directeur de thèse et responsable de l'équipe SBDM. Je le remercie pour sa confiance, son ouverture d'esprit et son esprit critique. Il n'est pas commun qu'un biologiste accepte un ingénieur en aéronautique dans son équipe et c'est sans hésitation qu'il a accepté ma candidature. Mes travaux s'inscrivent dans la continuité d'un projet et d'une vision développés par Olivier depuis plus de 20 ans. J'ai donc pu bénéficier d'un environnement propice aussi bien par la coopération avec des mathématiciens qu'avec des biologistes ouverts aux concepts de la biologie des systèmes.

Si j'ai énormément apprécié travailler avec Olivier, cela a aussi été le cas avec les autres membres de l'équipe. Sans données biologiques de qualité, toute analyse est inutile. C'est pourquoi je remercie Angélique, Anissa, Elodie et Catherine pour leur professionnalisme, leur bienveillance et nos multiples échanges sur des sujets aussi bien scientifiques qu'amicaux. J'associe également Sandrine, Geneviève et Gérard pour nos discussions sur des problématiques biologiques passionnantes, avec une mention spéciale à Gérard pour ces pauses conviviales à refaire le monde autour d'un café. Pour continuer sur des boissons plus éthyliques, je remercie Ronan toujours prêts à organiser des apéritifs (avec modération) au sein de l'équipe. Je remercie aussi Ulysse pour son apport essentiel à mes, ou plutôt nos, travaux. Il a su être patient et attentif avec un ingénieur parfois distrait sur la rigueur mathématique. Je salue également toutes les personnes qui sont passés dans l'équipe SBDM au cours de ces 3 années, en particulier François et Rebecca ainsi que les stagiaires Alice, Soorya, Diane, Emma et Kaushik. Cette équipe SBDM est la preuve que la pluridisciplinarité scientifique peut fonctionner dans une excellente ambiance et faire avancer la recherche. Pour autant, la fin de ma thèse n'est pas un adieu et j'espère travailler encore longtemps avec cette formidable équipe.

Cette thèse est le fruit d'une coopération entre l'équipe SBDM et Cosmo Tech, et sans la participation de Cosmo Tech elle n'aurait jamais pu avoir lieu. Je tiens donc tout particulièrement à remercier Cosmo Tech d'avoir accepté et financé ce projet. Je remercie aussi

Pierre-Alexis pour son accompagnement et son regard critique. De même je remercie Camilo pour nos nombreux échanges sur notre passion commune de la biologie des systèmes et de la langue espagnole. Il y a également tous les autres cosmonautes avec qui j'ai partagé d'excellents moments de convivialité. Malheureusement je ne peux pas tous les citer mais ils/elles se reconnaitront sans aucun doute.

Je remercie aussi l'équipe Dracula de l'INRIA à laquelle j'ai participée. Merci à Mostafa et Fabien pour l'animation de cette équipe, leurs retours sur mes travaux et leur sympathie. Je salue également tous les autres membres de l'équipe Dracula présents et passés.

Je remercie également le centre de calcul de l'IN2P3 du CNRS pour son professionnalisme et sa réactivité. En particulier je remercie Pascal Calvat qui a toujours été à mes côtés pour m'accompagner dans la découverte du calcul parallèle.

Pour continuer sur la thématique du calcul parallèle, je remercie aussi Eddy Caron et son équipe AVALON avec qui nous avons collaboré et qui m'ont beaucoup aidé.

Je remercie les membres de mon comité de suivi de thèse Grégory Batt et Christophe Arpin pour leur écoute et leurs précieux conseils.

De manière plus général, je remercie l'Etat qui m'a permis de reprendre mes études via un Congé Individuel de Formation pour le master, et qui a largement co-financé cette thèse via l'Agence National pour la Recherche et la Technologie (ANRT). Je mesure la chance d'avoir pu faire cette reconversion. J'espère que ces aides bénéficieront longtemps à toutes personnes curieuses désireuses de se reconvertir.

Enfin, Je remercie et embrasse très fort ma petite famille qui m'a suivie dans cette aventure Lyonnaise avec enthousiasme et dans la bonne humeur qui la caractérise. Ce projet de reconversion, qui ne se limite pas à la thèse, n'est pas seulement le mien mais aussi celui de ma famille, et en particulier de ma femme Sonia qui n'a pas hésité à me suivre loin de sa famille et de nos amis. Sa complicité et son sourire généreux m'ont permis de relativiser les petites difficultés et ont fait de ces 3 années des belles pages de notre histoire. Pour tout cela je lui dédie cette thèse (même s'il semble qu'elle n'ait pas tout compris). A noter aussi que notre famille s'est agrandie pendant cette thèse avec l'arrivée de Bastien qui a rejoint ses frères Esteban et Rafael pour le plus grand plaisir de ses parents et grands-parents qui nous ont bien aidé à veiller sur eux et que je remercie également.

Table des matières

Contexte et objectif de la thèse CIFRE

Ce projet de thèse CIFRE est la rencontre entre 3 acteurs ; Cosmo Tech, Olivier Gandrillon et moi-même. Cosmo Tech est une société technologique spécialisée dans la modélisation des systèmes complexes. L'équipe "Systems Biology of Decision Making" (SBDM) est dirigée par Olivier Gandrillon au sein du "Laboratoire de Biologie et Modélisation de la Cellule" (LBMC) de l'ENS de Lyon. Elle a pour objectif de comprendre la prise de décision des cellules dans le cadre de processus de différenciation. Ces deux acteurs ont déjà collaboré pour l'étude de problématiques biologiques à l'aide d'approches mécanistes. Dans le cadre de l'ANR Predivac, Cosmo Tech a produit en collaboration avec l'EPI INRIA Dracula, dont fait partie Olivier Gandrillon, un modèle de la réponse immunitaire primaire T CD8 [1]. Ce modèle est multi-échelles : il décrit des échelles moléculaires et cellulaires couplées, et permet d'observer de manière quantitative l'émergence de phénomènes populationnels complexe dont la dynamique dépend de ce couplage. Une des limitations majeures à la construction de tels modèles reste l'identification des interactions moléculaires, basée en pratique sur une lecture aussi exhaustive que possible de la littérature et de discussions avec des spécialistes du domaine. Il s'agit d'un processus chronophage qui nécessite que la plupart des connaissances ait déjà été acquises et publiées. Il n'existe en pratique aucun algorithme encore capable d'inférer ces interactions à partir de l'analyse de données.

Cette thèse s'inscrit également dans mon projet personnel de reconversion. Fort d'une expérience de 10 ans en ingénierie sur le développement de pilotes automatiques d'avion, mon objectif est d'appliquer les outils, approches et concepts du domaine de l'aéronautique aux problématiques biologiques. L'idée sous-jacente, que je ne développerai pas ici, est qu'il existe des similitudes entre l'architecture des systèmes biologiques et celle des systèmes critiques développés dans l'industrie. Paradoxalement, s'il est convenu que les systèmes biologiques sont largement plus complexes que les systèmes créés par l'Homme, la biologie a très peu recours aux outils et moyens développés dans l'ingénierie pour maitriser les systèmes complexes.

Cette thèse a donc pour objectif commun de construire un outil informatique inspiré de l'ingénierie permettant d'automatiser la tâche d'inférence de réseau moléculaire dynamique, en l'occurrence de réseaux de régulation de gènes, à partir d'un nouveau type de données, les données transcriptomiques acquises à l'échelle de la cellule unique. Cette méthode devra

surmonter les limites actuelles de l'inférence de réseaux de régulation de gènes.

1 Introduction

1.1 Inférence de réseau de régulation de gènes (RRG) à partir de données d'expression

On attribue aux systèmes vivants, et en particulier les cellules, la capacité de "prise de décision" pour modifier leur comportement suite aux variations de leur environnement. Cette propriété est essentielle pour le bon fonctionnement d'un organisme multicellulaire ou unicellulaire. Le comportement d'une cellule peut se définir par l'adaptation de ces capacités fonctionnelles suite aux modifications de son environnement ou de son état interne. Ces fonctions étant en grande partie réalisées par des protéines, il convient donc pour la cellule de réguler l'expression des gènes codant pour les protéines appropriées. C'est ce qui est réalisé par la cellule dans le cas d'une différenciation. Cette régulation se fait en partie par une série d'interactions entre gènes qui s'activent ou s'inhibent. La connaissance de ces réseaux de régulation de gènes (RRG) permettrait donc en théorie de comprendre, de prédire, et potentiellement d'influencer le comportement d'une cellule et ainsi proposer des nouveaux traitements dans le cas de pathologies impliquant une dérégulation des RRG. Beaucoup de travaux ont été menés depuis 20 ans pour répondre à ce défi, et malgré quelques réussites, le problème d'inférence des RRG reste largement ouvert. Toutes ces approches ont en commun l'analyse de données d'expression à l'aide de modèles statistiques ou mécanistes de RRG afin d'inférer les relations de corrélation ou de causalité.

1.2 RRG et différenciation

Dans cette partie nous présentons les mécanismes et les rôles des RRG avant de détailler le cas précis de la différenciation. Après un aperçu de la vision classique du processus de différenciation, nous verrons comment les RRG sont impliqués dans une vision plus moderne.

1.2.1 Les RRG coordonnent l'expression des gènes

1.2.1.1 Le contrôle de l'expression génétique

Toutes les cellules d'un organisme multicellulaire contiennent le même code génétique contenu dans leurs molécules d'ADN (à de rares exceptions près comme des cellules immunitaires ou les gamètes), et pourtant elles présentent des phénotypes extrêmement variés. Cette propriété s'explique par le contrôle différentiel des gènes exprimés. Il est utile à cette étape de définir plus précisément la notion d'"expression d'un gène" qui sera utilisée dans ce manuscrit. Nous supposons qu'un gène correspond à une séquence d'ADN codante pour une protéine. Nous excluons de fait tous les gènes qui ne produisent pas des ARNm, bien que les ARN non codant jouent aussi un rôle important comme il a été montré pour les miARN, siARN, lncARN, etc [2, 3]. Lorsqu'un gène est exprimé on suppose que son ARNm est transcrit de façon détectable et significative. Nous convenons toutefois que cette définition est très inappropriée à l'échelle de la cellule unique à cause de la stochasticité de l'expression comme nous le verrons plus tard.

Une cellule peut donc contrôler l'expression de ces gènes et ainsi produire des protéines nécessaires à son fonctionnement. On estime qu'à tout moment une cellule humaine exprime entre 30% et 60% de ces 25 000 gènes [4]. Cependant, beaucoup de processus sont communs à tous les types de cellules, ainsi seules une partie des gènes sont spécifiques à un type cellulaire. Par exemple, on retrouve les protéines de la chromatine (histones), les ARN polymérases, des enzymes importantes du métabolisme ou encore des protéines du cytosquelette dans toutes les cellules, même si leur niveau d'expression peut varier. En revanche, l'hémoglobine n'est détectable que dans les érythrocytes, elle est donc spécifique à ce type cellulaire et participe à sa définition.

Le contrôle de l'expression des gènes chez les eucaryotes peut se faire à différents niveaux allant de l'ADN à la protéine. Le niveau le plus en amont et le plus étudié est celui de la transcription. La transcription de l'ARNm à partir de l'ADN implique une série d'acteurs, comme les facteurs de transcription généraux et spécifiques, qui doivent se coordonner dans l'espace et le temps pour amorcer et entretenir le processus de transcription. Mais la régulation de l'expression peut aussi se faire plus en aval lors de la maturation de l'ARNm, de

son transport dans le cytoplasme, de sa traduction ou de sa dégradation [5]. Au niveau de la protéine il peut aussi y avoir des régulations dites post-traductionnelles qui altèrent son activité ainsi que sa stabilité [6, 7]. A ces différences de niveaux de contrôle, il faut rajouter les différences de dynamique. Il existe des contrôles stables et persistants, comme dans certains cas de régulations épigénétiques qui induisent du "silencing" de chromosome [8], contrairement aux contrôles plus dynamiques par facteurs de transcription qui exigent que les protéines régulatrices soient présentes en permanence.

Bien que la plupart des études sur l'inférence des RRG se limitent à la régulation transcriptionnelle, nous attirons l'attention sur le fait que dans ce manuscrit nous considérons plusieurs niveaux de régulation. Cependant nous ne considèrerons pas les régulations de type persistante comme les mécanismes épigénétiques.

1.2.1.2 Définition des RRG

Lorsqu'une cellule modifie l'expression de ces gènes suite à un changement d'environnement ou d'un stimulus, il est très rare qu'un seul gène soit impacté. C'est un ensemble de gènes fonctionnellement liés qui est régulé comme dans le cas des cellules hépatiques qui répondent aux glucocorticoïdes en sur-exprimant une série de protéines spécialisées dans la production de glucose [9]. Il y a donc une coordination du contrôle de l'expression des gènes qui est-elle même dépendante des gènes exprimés. En effet, seules quelques types cellulaires sont capables de répondre aux glucocorticoïdes contrairement aux autres types cellulaires aussi exposés. On parle alors de RRG comme l'ensemble des interactions des gènes qui contrôlent l'expression d'autres gènes à tous les niveaux (transcription, traduction, etc). Nous allons maintenant détailler des concepts importants de structures et d'états des RRG pour les définir précisément.

On nomme "structure" du RRG l'ensemble des interactions possibles entre gènes. On nomme "état", pour une cellule donnée à un instant T, l'ensemble des niveaux d'expression des ARNm et protéines de tous les gènes. La structure d'un RRG est constante dans le temps et ne dépend pas du type cellulaire, contrairement à l'état d'une cellule. C'est cette différence fondamentale qui définit ces 2 notions. La structure d'un RRG inclus les conditions pour qu'une interaction soit effective. Dans l'exemple précédent, pour que les gènes cibles des glucocorticoïdes soient induits, il faut d'une part la présence de l'hormone, et d'autre

part la présence de protéines spécifiques aux cellules hépatiques qui autorisent la réponse des gènes cibles. Dans le cas d'une cellule de peau, qui a la même structure de RRG pour la réponse aux glucocorticoïdes que la cellule hépatique, son état est différent et n'autorise pas l'activation de cette interaction.

On peut alors définir la tâche d'inférence de RRG comme l'identification de la structure du RRG à partir de mesures expérimentales de l'état du RRG dans différentes conditions ou lors d'une cinétique. Il est utile de préciser ici que seule une petite partie de la structure du RRG sera explicite, et qu'une partie importante sera cachée comme illustré dans la figure 1. Pour comprendre cette limite reprenons l'exemple des cellules hépatiques. Si on cherche à retrouver la série d'interactions qui mènent à l'activation des gènes cibles suite à la présence de l'hormone, il ne sera certainement pas possible de connaitre toutes les conditions nécessaires pour que ce RRG soit opérationnel. Si un gène A active un gène B, il est fort probable que cette interaction nécessite la présence de gènes constamment et spécifiquement exprimés dans ce type cellulaire. La notion de RRG est donc relative et restreinte aux gènes dont l'expression varie au cours du processus étudié.

1.2.1.3 Approximation des interactions géniques par la séparation des échelles de temps

Nous allons maintenant définir plus précisément la notion d'interaction entre gènes qui est en soit une abstraction et cache des mécanismes complexes. Comme citée plus tôt dans cette introduction, la régulation de l'expression des gènes peut se faire à de multiples niveaux. Chaque niveau contribue à la régulation, ce qui *a priori* complexifie la régulation globale du RRG. Cependant, nous allons voir qu'en appliquant une hypothèse de séparation des échelles de temps, on peut ramener la régulation globale du RRG à ses processus les plus lents, que nous supposerons être la transcription et la traduction.

Dans notre étude on supposera que les interactions entre gènes se font *via* les protéines. De fait, nous négligeons délibérément et par souci de simplification la régulation effectuée par les ARN non-codants, notamment au niveau de la dégradation. Une fois que la protéine régulatrice est produite dans le cytoplasme, elle peut subir des cascades de modifications post-traductionnelle, comme des phosphorylations. Elle peut aussi induire à son tour une

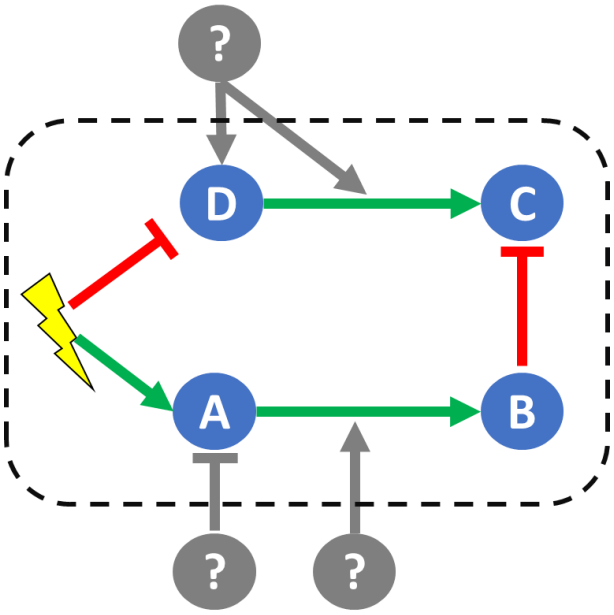


Figure 1 – Représentation schématique d'un RRG
Ce schéma est un exemple de RRG faisant intervenir 4 gènes A, B, C et D régulés suite à un stimulus (flash jaune). Les interactions entre stimulus/gène ou gène/gène sont représentées par des flèches vertes (activation) ou des "blocs" rouges (inhibition). Ces interactions constituent la structure visible du RRG. Les gènes peuvent aussi être influencés par d'autres gènes non identifiés (gris avec " ?") mais non impactés par le stimulus et donc constants au cours de la stimulation. Ces gènes constants peuvent aussi réguler les interactions comme pour l'interaction entre le gène A et B. Ces interactions définissent la structure cachée du RRG. Le RRG sera défini comme l'ensemble de la structure visible dans le bloc en pointillé. Il faudra cependant garder à l'esprit que ce RRG dépend fortement du contexte cellulaire. L'état du RRG sera défini par le niveau des ARNm et protéines des gènes A,B,C,D au cours de la stimulation.

cascade de phosphorylation sur d'autres protéines qui *in fine* vont aboutir à l'activation d'un facteur de transcription qui va réguler l'expression d'un gène cible comme montré dans la figure 2. Nous faisons l'hypothèse que les temps caractéristiques des réactions qui ont lieu dans le cytoplasme sont très courts par rapport à la dynamique de production/dégradation des ARN et protéine qui sont de l'ordre de plusieurs heures. De fait, une fois la protéine régulatrice produite, toutes ces réactions intermédiaires, qui agissent aussi comme des filtres, atteignent très vite leur régime stationnaire et le transfert global se limite au gain statique engendré par ces réactions.

Nous allons donc nous concentrer sur l'étude des dynamiques lentes du RRG et négliger

les plus rapides. Cette hypothèse de séparation des échelles de temps est justifiée par les points suivants. Les processus de différenciation durent généralement plusieurs jours, ce qui sous-tend une dynamique équivalente pour que le RRG se stabilise. L'autre raison est liée aux contraintes expérimentales. Les techniques actuelles de mesures des observables du système des RRG se limitent aux ARN et protéines. Étant donné la dynamique de ces molécules, il est inutile de les mesurer toutes les minutes. Enfin, le nombre de points expérimentaux est en pratique limité par leur coût.

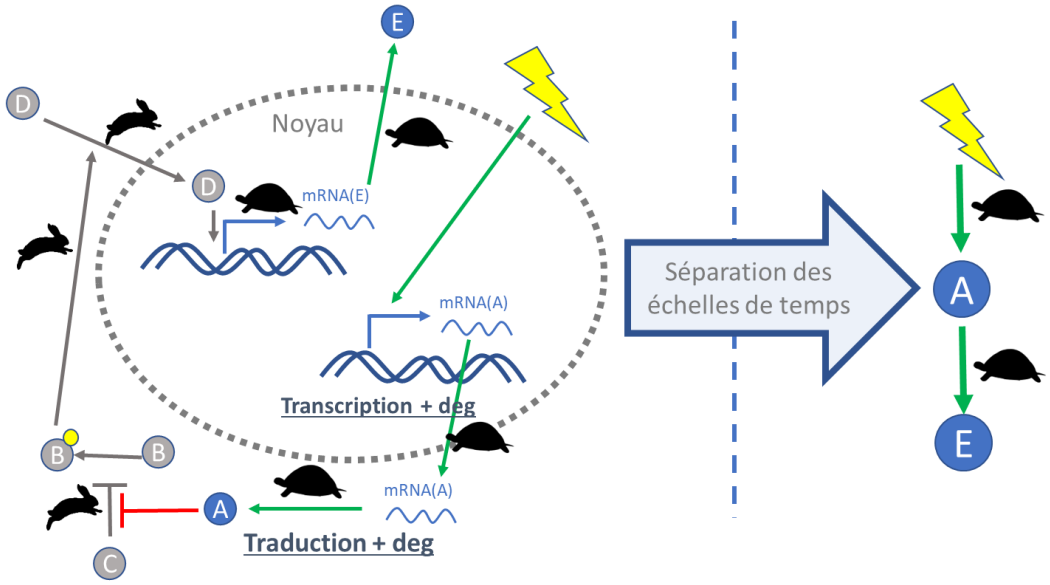


Figure 2 – Séparation des échelles de temps
La partie gauche du schéma représente un RRG impliquant les gènes A, B, C, D, E. Seuls les gènes A et E subissent une régulation de leur expression. A est directement sous le contrôle d'un stimulus. Le gène D est un facteur de transcription séquestré dans le cytoplasme qui peut être relâché dans le noyau grâce à l'effet de la protéine B phosphorylée indirectement par l'action de A qui inhibe la protéine C qui empêche la phosphorylation de B. Les processus de transcription et de traduction sont typiquement lents (en heures, représentés par les tortues) comparés aux autres processus de phosphorylation ou de translocation (en secondes ou minutes, représentés par les lapins). L'approximation de séparation des échelles de temps (à droite) ne considère que les processus lents et permet de réduire cette cascade de réaction à l'activation de A par le stimulus puis l'activation de E par A.

Les RRG interviennent dans de nombreux processus de réponse à un stimulus, comme nous venons de le voir. Il peuvent aussi servir de circuit logique pour traiter l'information [10], ou être utilisé en oscillateur comme pour l'horloge circadienne [11]. Mais l'exemple le

plus courant est l'implication des RRG dans les processus de différenciation, qui sera notre modèle biologique dans le cadre de cette thèse, et que nous allons introduire dans la partie suivante.

1.2.2 Le processus de différenciation

Les RRG ont historiquement été étudiés chez les eucaryotes dans le cas du développement et sont donc fortement liés au processus de différenciation. Nous allons dans un premier temps présenter le processus de différenciation dans sa vision classique, puis nous détaillerons le cas particulier de la différenciation érythrocytaire qui est le modèle étudié dans notre équipe, et enfin nous étudierons le lien avec les RRG et la nouvelle vision qui en découle.

1.2.2.1 La vision historique du processus de différenciation cellulaire

La différenciation est le processus par lequel une cellule va changer de type cellulaire pour se spécialiser. La différenciation intervient au cours du développement pour créer un organisme à partir d'une cellule unique, l'œuf, ainsi que dans la vie adulte pour régénérer des tissus. La différenciation intervient généralement au cours d'une division cellulaire qui peut être symétrique ou asymétrique, ce qui implique respectivement que les cellules filles sont différenciées et identiques ou qu'une cellule fille est identique à la mère et l'autre différenciée. La différenciation est dans la vision classique un processus irréversible qui conduit une cellule immature à se spécialiser vers une cellule plus mature. A la dernière étape de différenciation la cellule est complètement mature et ne peut ni se diviser ni se différencier. Cette irréversibilité nécessite la mémorisation par la cellule des étapes de différenciation par lesquelles elle est passée. Les différentes étapes de la différenciation à partir de l'œuf peuvent alors se voir comme un arbre généalogique avec des liens de descendance dirigés. Un type cellulaire peut engendrer plusieurs types cellulaires, on parle alors de multi-lignage. Cette vision est parfaitement déterministe et fait généralement intervenir des gènes dits "maîtres" dont la seule activation au cours d'une différenciation permet l'activation, ou l'inhibition, de tous ses gènes cibles spécifiques au nouveau type cellulaire [12]. Ces gènes maîtres sont très souvent des facteurs de transcription ou des suppresseurs de tumeur. Cependant cette vision classique est aujourd'hui largement remise en cause comme nous allons le voir par la suite.

1.2.2.2 Caractéristiques du processus de différenciation érythrocytaire

Le modèle biologique que nous étudierons dans ce manuscrit concerne l'hématopoïèse, et plus particulièrement l'érythropoïèse, chez le poulet [13]. L'hématopoïèse consiste en la génération de toutes les cellules sanguines à partir d'un seul type de cellule souche sanguine par un processus de différenciation hierarchique qui comporte 2 branches principales lymphoïde et myéloïde [14]. Les progéniteurs myéloïde vont progressivement se différencier et donner lieu, entre autres, à des progéniteurs érythrocytaires qui *in fine* vont uniquement se différencier en érythrocyte. Les érythrocytes, communément appelés globules rouges, sont des cellules spécialisées du sang qui servent à transporter l'oxygène dans tous les organes *via* le réseau sanguin. L'oxygène est fixé par l'hémoglobine, une protéine exclusivement et abondamment produite dans ce type cellulaire. Au dernier stade de la différenciation érythrocytaire le noyau est expulsé et les mitochondries sont éliminées par autophagie [14]. Cette étape drastique de différenciation irréversible peut se comprendre dans l'intérêt pour la cellule de n'être plus qu'un "sac" à hémoglobine, les mitochondries qui consomment l'oxygène diminueraient l'efficacité du transport d'oxygène. Cependant, ces caractéristiques ne s'appliquent pas chez les oiseaux. Le noyau est conservé mais condensé à un tel niveau qu'il est non fonctionnel et toute activité de transcription est indétectable. Les mitochondries ne sont pas phagocytées et restent fonctionnelles. Les progéniteurs érythrocytaire T2EC [13] que nous étudions sont des cellules de culture primaire issus d'embryon de poulet. Elles peuvent être maintenues en état d'auto-renouvellement ou induites en différenciation en 4 ou 5 jours.

1.2.2.3 La différenciation et les RRG dans la vision moderne

La régulation de l'expression des gènes a initialement été étudié chez les bactéries [15]. Chez les eucaryotes, ou la régulation est plus complexe, le contrôle de l'expression a été étudié initialement dans le développement de l'oursin et de la drosophile [4]. Cela a permis d'établir des RRG complexes sous forme de circuits logiques de commutateurs génétiques inter-connectés [16] expliquant de façon mécaniste les propriétés de la différenciation. Ainsi, les gènes "maîtres" sont activés séquentiellement en fonction des signaux extérieurs. De plus, les boucles positives présentent dans les RRG permettent d'expliquer la notion de mémoire nécessaire pour l'irréversibilité du processus. Dans cette vision, le processus de différenciation

est la phase transitoire d'un RRG qui passe d'un état stable à un autre suite à une stimulation.

Cependant, nous savons aujourd'hui que le dogme du processus de différenciation hiérarchique irréversible ne tient plus suite aux expériences de l'équipe de Yamanaka sur la reprogrammation des cellules différenciées [17]. En effet, on sait aujourd'hui reprogrammer de nombreuses cellules différenciées en cellules pluripotentes [18] capable de générer tous les lignages. Il existe aussi de nombreux exemples de transdifférenciation [19] sans passer par l'étape IPS. Le cadre proposé par les RRG, avec la notion de structure invariante et d'état variable, est parfaitement compatible avec ces observations et permet de proposer des mécanismes pour les expliquer.

Malgré les succès des RRG pour expliquer les processus complexes de différenciation, certaines observations comme la différenciation spontanée ou des cellules rebelles [20] ne sont pas expliquées par la vision déterministe de circuits logiques. Comme nous le verrons plus tard dans cette introduction, des mesures en cellules uniques ont mis en évidence l'extrême stochasticité de l'expression génétique. Les RRG apparaissent alors plus comme un réseau de contraintes qui canalisent la stochasticité de l'expression génétique. Si la contrainte exercée par le RRG est très forte, la probabilité associée à un état final est très importante et la réponse de la cellule peut sembler déterministe. Au contraire, si les contraintes sont plus lâches et qu'il existe plusieurs états stables, la cellule peut alors explorer de façon aléatoire cet espace et basculer dans un état ou un autre [21].

Nous allons dans la suite présenter différentes méthodes d'inférence des RRG qui pour la majorité se basent sur la vision classique de la différenciation. Cependant, nous verrons plus tard comment la nouvelle vision peut être utile à l'inférence des RRG à partir de mesures en cellule unique.

1.3 Inférence des RRG à partir de données en population

L'idée d'inférer des RRG à partir de données d'expression a commencé à la fin des années 1990 avec l'arrivée des premières puces d'expression à ADN [22, 23]. Les algorithmes d'inférence ont ensuite suivi l'évolution des technologies de transcriptomique jusqu'aux toutes dernières données en cellule unique. Afin d'avoir une image la plus complète et précise possible sur l'état de l'art, il est préférable de commencer par présenter les différentes familles d'al-

gorithmes développés pour l'analyse de données en population, qui ont ensuite été adaptées pour l'analyse des données en cellule unique.

1.3.1 Les données d'expression en population

La technologie des puces à ADN a permis pour la première fois de mesurer le niveau d'expression à l'échelle du génome d'une population de cellules. De fait, plutôt que de s'intéresser à un groupe de gènes en particulier, on pouvait alors observer l'ensemble des gènes impactés par une stimulation. Les biologistes ont alors compris le besoin d'étudier la réponse des cellules à un niveau systémique plutôt que réductionniste. La notion de réseau génétique à alors pris toute son importance. Les algorithmes d'inférence de réseau se sont donc développés à ce moment, les approches classiques d'analyse manuelle des données d'expression ne pouvant plus se faire sur des milliers de gènes.

Les données d'expression sont classées en 2 grandes catégories, les perturbations et les cinétiques. Le principe des expériences de perturbation est d'imposer une sous-expression ou sur-expression de certains gènes du RRG et de mesurer l'impact sur les autres gènes engendrés par les interactions. L'idée est de retrouver les interactions par "retro-engineering" des mesures. En pratique ces expériences se font plus en majorité chez des organismes unicellulaires comme la bactérie E.Coli [24] ou la levure S.cerevisiae [25] facilement manipulables génétiquement. Cette technique permet de réaliser des criblages à haut débit, comme par exemple dans [25] ou 300 perturbations ont été réalisés. On trouve aussi ce type d'expérience dans des cellules humaines [26].

Les données de cinétique permettent de suivre l'évolution dynamique de l'expression des gènes au cours de processus physiologique, ou suite à une perturbation. L'idée sous-jacente est que la structure du réseau impose la dynamique observée. Ainsi, des levures ont été synchronisées [27] pour faire apparaitre les oscillations dues au cycle cellulaire dans l'expression de 800 gènes (soit un tiers du génome) avec l'objectif de retrouver le RRG impliqué dans le cycle cellulaire. Chez la mouche *Drosophila melanogaster* [28] 4028 gènes ont été mesurés à 66 stades séquentiels du développement de la fertilisation au stade adulte.

Les données d'expression systémiques peuvent aussi servir à des fins de classification comme dans le cas des tumeurs [29]. L'objectif est d'établir des profils d'expression et de

les corréler avec le phénotype tumoral. Beaucoup d'autres données ont été produites, et afin de faciliter leur accès et utilisation, des banques de dépôt informatique ont été créées comme Gene Expression Omnibus du NCBI pour héberger des milliers de données de puces d'expression. Les données de séquençage d'ARN sont arrivées plus tard et ont finalement été peu utilisées dans l'inférence des RRG [30]. Toutes les approches présentées par la suite n'utilisent que des données de puces.

1.3.2 Les approches Bayésiennes

Les données d'expression des puces sont des prises instantanées ("snapshot") du niveau expression de plusieurs centaines de gènes et elles sont considérées comme bruitées et semi-quantitatives. Les approches Bayésiennes, qui modélisent les RRG par un processus stochastique, sont particulièrement adaptées à l'analyse de ces données et ont été utilisées très tôt [31].

1.3.2.1 Les réseaux Bayésiens

Dans un réseau Bayésien le RRG est représenté par un graphe acyclique dirigé noté G (figure 3) ou les nœuds sont des variables aléatoires notées X_i qui sont fonction de l'état des parents notés $Pa(X_i)$. Une interaction dirigée dans le graphe représente la dépendance du fils par rapport à ses parents, qui se traduit par une distribution conditionnelle de l'état du fils en fonction de l'état des parents. Dans l'hypothèse Markovienne cette relation revient à dire que le fils est indépendant de ses non-descendants étant donnés ses parents directs. C'est cette relation qui est au cœur de la relation de causalité dans un réseau Bayésien. L'ensemble du réseau peut être défini par une distribution jointe entre toutes les variables du réseau. En utilisant le théorème de Bayes, associé à l'hypothèse Markovienne d'indépendance conditionnelle par rapport aux parents, on peut réduire l'écriture de cette distribution jointe sous la forme suivante :

$$P(X_1, ..., X_n) = \prod_{i=1}^{n} P(X_i|Pa(X_i)) \quad (1)$$

Le graphe G permet d'identifier les parents de chaque nœud, mais pour définir complètement la distribution jointe du réseau Bayésien il faut spécifier chaque probabilité condi-

tionnelle $P(X_i|Pa(X_i))$. Ces probabilités conditionnelles peuvent être définies à l'aide de fonctions discrètes [32, 33, 34, 35] ou continues (souvent des Gaussiennes) qui sont spécifiées par un jeu de paramètres noté Θ. Un réseau Bayésien est donc défini par son graphe G et ses paramètres Θ.

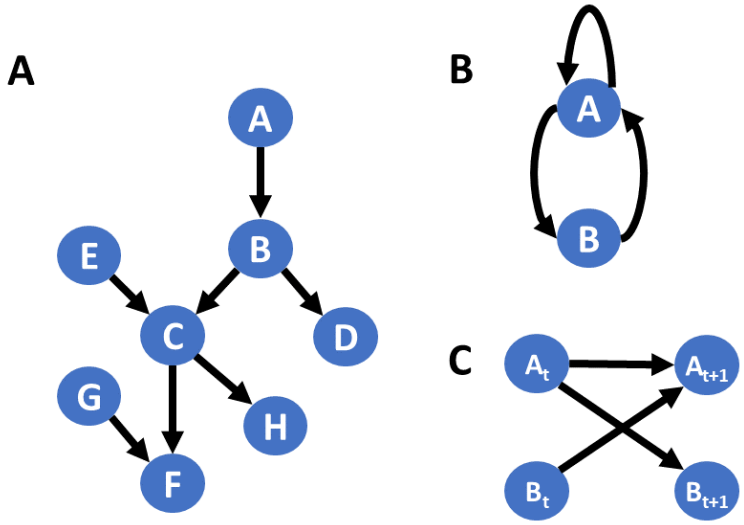


Figure 3 – Réseaux Bayésiens
A) Un réseau Bayésien est représenté par un graphe acyclique dirigé. Un nœud représente un gène définit par une variable aléatoire dont la densité de distribution dépend de ses régulateurs, ou parents. Dans cet exemple le gène C a une distribution qui est liée à celle de ses parents E et B, mais aussi à son grand parent A, son cousin D et ses descendants F et H. La distribution de C est indépendante à G car ils n'ont aucun lien de parenté. Avec l'hypothèse Markovienne, la distribution de C conditionnée à tous les gènes sauf ses descendants revient à la distribution de C conditionnée à ses parents. On a donc $P(C|A, B, D, E, G) = P(C|B, E)$. B) et C) Le graphe en B) illustre une structure de graphe qui comportant des boucles qui ne peut pas être représentée par un graphe acyclique dirigé. Le graphe C) représente le réseau B) par un réseau Bayésien Dynamique à l'aide d'un graphe acyclique. Les nœuds A et B sont représentés à l'instant t et $t+1$. Une interaction se fait toujours de t à $t+1$ ce qui garantie l'absence de boucle dans ce graphe acyclique.

1.3.2.2 L'inférence Bayésienne

Pour inférer le RRG à partir d'un jeu de donnée D l'idée est de trouver le graphe G qui explique le mieux D. Pour cela on définit généralement la probabilité postérieure $P(G|D)$ d'avoir un graphe G connaissant D et on cherche le graphe G qui maximise cette probabilité.

On peut utiliser une fois de plus le théorème de Bayes pour décomposer $P(G|D)$ qui peut s'écrire sous la forme :

$$P(G|D) = \frac{P(D|G)P(G)}{P(D)} \tag{2}$$

Le terme $P(D|G)$ est appelé la vraisemblance et $P(G)$ l'antérieur. $P(D)$ correspond ici à une constante. Pour calculer la vraisemblance on utilise souvent les scores Bayesian Information Criteria (BIC) ou Bayesian Dirichlet equivalence (BDe). Ces scores intègrent notamment la complexité des graphes pour éviter les problèmes de sur-paramétrisation (over-fitting). L'inférence se fait en 2 temps, on propose d'abord une structure de graphe G à tester, puis on cherche le jeu de paramètre Θ le plus approprié pour reproduire les données. C'est à cette dernière étape que l'on associe un score au réseau candidat. La recherche des graphes les plus représentatifs dans la première étape est un problème très difficile de type NP-complet qui impose l'utilisation d'algorithmes de recherche heuristique comme "Markov Chain Monte Carlo" (MCMC) ou "greedy-hill climbing" [36, 37]. Le problème d'inférence étant indéterminé vu le faible nombre de donnée par rapport à la complexité du problème, plusieurs réseaux sont souvent retournés. Le terme antérieur $P(G)$ peut être utilisé pour intégrer des connaissances *a priori* sur les gènes issus de la littérature ou des bases de données d'annotation des gènes [38, 39] ou par regroupement en "cluster" notamment en utilisant des corrélations temporelle [39, 40]. Des hypothèses de "rareté" des interactions sont également parfois utilisé pour réduire le nombre d'interaction possibles [41, 34].

1.3.2.3 Les réseaux Bayésiens Dynamiques

Comme nous le verrons dans le prochain paragraphe, une des limites majeures des réseaux Bayésiens est qu'ils sont statiques et n'autorisent pas les boucles dans les RRG. Une alternative est possible avec les réseaux Bayésiens Dynamiques [33, 37, 42, 34, 40, 35] qui sont une variante des réseaux Bayésiens. L'idée est de dupliquer le graphe avec un graphe G_t et un autre graphe G_{t+1} qui représentent respectivement l'influence du réseau au temps t sur le réseau au temps $t+1$. De fait on introduit la notion de dynamique, puisque l'état du réseau passé impacte l'état présent, et on autorise les boucles comme l'illustre la Fig 3-B/C. La méthode d'inférence reste inchangée et autorise l'utilisation de données temporelles discrètes

qui seront utilisées 2 à 2 pour 2 temps consécutifs. On retrouve dans cette formulation l'idée que l'information met un certain temps à se propager dans le réseau.

1.3.2.4 Avantages et limites

Les approches Bayésiennes présentent l'avantage d'identifier des causalités selon le principe de l'indépendance des probabilités et elles sont compatibles avec l'hypothèse de séparation des échelles de temps. Les modèles Bayésiens comportent peu de paramètres ce qui est un avantage en terme d'identifiabilité et de puissance statistique nécessaire. Ces méthodes sont aussi robustes aux données bruitées.

Cependant, en pratique il reste assez difficile d'identifier le gène régulateur et le gène régulé, même si l'utilisation de réseaux Bayésiens dynamiques facilite l'exercice. La discrétisation des données dégrade quantitativement la précision de l'approche et ajoute de nombreux paramètres en fonction du nombre d'état discret. Une solution est le passage en continu avec des hypothèses linéaires Gaussiennes qui simplifient le modèle. La combinatoire devient très vite importante (problème de type NP-hard) ce qui limite l'inférence aux petits réseaux typiquement inférieurs à 20 gènes [33, 37, 42, 34]. D'où le recours à des méthodes heuristiques et des hypothèses arbitraires de rareté d'interactions. L'utilisation de connaissances *a priori* est pratiquée pour contourner ces difficultés, mais bien souvent il s'agit de co-clustering à partir de base de donnée d'annotation de gène dont l'utilisation directe est discutable. Les méthodes Bayesiennes non dynamiques sont en pratique trop limitées car elles ne considèrent pas les évolutions temporelles et ne peuvent pas inférer des boucles d'interactions.

1.3.3 Les approches par la théorie de l'information

Dans la famille des approches statistiques on trouve également celles basées sur la théorie de l'information. Le principe repose sur la corrélation : si 2 gènes partagent de l'information il existe certainement une interaction plus ou moins directe. Comme dans le cas Bayésien, cette approche s'appuie sur la recherche de dépendance statistique, mais dans ce cas la nature des relations inférées entre gènes est limitée a de la corrélation. Il est de fait impossible d'identifier des liens de causalité contrairement aux approches Bayésiennes. Cependant, l'efficacité de

cette méthode permet de traiter des gros volumes de données pour analyser des données à l'échelle du génome contrairement à son homologue Bayésien.

1.3.3.1 L'information mutuelle

L'information mutuelle $IM_{i,j}$ entre 2 gènes i et j est une généralisation de la corrélation. Elle se base sur l'entropie au sens de Shannon, noté H_i pour le gène i, et est définit de la façon suivante :

$$IM_{i,j} = H_i + H_j - H_{i,j} \tag{3}$$

L'entropie est donnée par la formule suivante dans le cas où la mesure de l'expression d'un gène peut prendre n valeurs finies $(x_1, ..., x_n)$ avec les probabilités associées $(p(x_1), ..., p(x_n)$:

$$H_i = -\sum_{k=1}^{n} p(x_k) log2(p(x_k)) \tag{4}$$

Si $IM_{i,j} = 0$ cela signifie que la distribution jointe des 2 gènes ne contient pas plus d'information que les 2 gènes pris séparément. Si $IM_{i,j} > 0$ cela signifie que les 2 gènes sont liés, leur distribution jointe contient moins d'information que les gènes pris séparément. Une part de l'information entre les 2 gènes est redondante. L'idée est donc que plus l'information mutuelle entre 2 gènes est importante, plus la probabilité d'une interaction directe est importante.

1.3.3.2 Principe de l'inférence

La première application de la théorie de l'information pour inférer des RRG a été décrite par [43] et appliqué sur des données génomique chez la levure. Le principe est de calculer pour toutes les paires de gènes l'information mutuelle. Il existe plusieurs méthodes [44] pour cela. Puis elles sont comparées à un seuil fixe et seules les interactions avec une information mutuelle supérieure au seuil sont conservées. On notera que cette méthode exige que les données soient statistiquement indépendantes, dans le cas contraire des corrélations expérimentales seraient interprétées à tort. C'est pourquoi on utilise généralement des données en état stationnaire [43, 26, 45].

L'algorithme le plus populaire utilisant l'information mutuelle est ARACNE [26, 45]. Il utilise la méthode des noyaux Gaussiens pour calculer l'information mutuelle. Il utilise également le "Data Processing Inequality" pour éviter d'identifier les interactions indirectes. Prenons l'exemple de 3 gènes A, B et C en cascade avec les interactions A->B->C. Il existe une corrélation entre A et C et donc $M_{A,C} > 0$. Cependant l'idée est que la corrélation issue d'une interaction indirecte est plus faible que les interactions directes et de fait on a l'inégalité $M_{A,C} < min(M_{A,B}, M_{B,C})$. En comparant les $IM_{i,j}$ entre eux on peut alors éliminer les interactions indirectes.

1.3.3.3 Avantages et limites

Le principal avantage de ces méthodes par rapport aux autres est le traitement de gros volumes de données. Les larges RRG inférés peuvent ensuite être analysés pour identifier des "hub", ce qui peut être pratique pour la recherche médicale [26]. Toutefois, la capacité prédictive de ces RRG est limitée car il est difficile de prédire les conséquences en cas de perturbation des hubs. Une autre limite importante à considérer est l'impossibilité de cette méthode à inférer les boucles de rétro-contrôle qui sont supposées présentes en nombre et jouent un rôle important dans les RRG. Enfin ces méthodes font appel à des hypothèses linéaires Gaussiennes qui impactent la pertinence biologique des RRG inférés. En pratique l'information mutuelle et une simple corrélation linéaire de Pearson donnent environ les mêmes résultats [44].

1.3.4 Les approches Booléennes

L'utilisation de réseaux Booléens pour représenter les RRG, et même les réseaux métaboliques, date de 1969 [46]. Il est intéressant de noter que dans cette étude théorique, l'objectif n'est pas de retrouver des interactions au sein d'un RRG, mais d'étudier les propriétés émergentes de ces réseaux construits aléatoirement comme l'aurait fait l'évolution. Kauffman a montré que ces réseaux Booléens aléatoires étaient stables, possédaient des modes cycliques cohérents avec le cycle cellulaire ou bien encore présentaient des états multi-stationnaires comparables aux différents types cellulaires. Cette étude illustre parfaitement l'intérêt de

modèles dits mécanistes, en opposition aux modèles statistiques que nous avons décrits jusqu'à présent.

1.3.4.1 Les réseaux Booléens

Un réseau Booléen est constitué d'un ensemble de variables $(x_1, ..., x_n)$ qui représentent les gènes et qui peuvent prendre les valeurs $(0, 1)$ représentant respectivement l'état inactif ou actif. Un gène est donc vu comme un commutateur. On retrouve ici l'idée d'un parallèle fort entre circuit biologique et circuit informatique. L'état d'une variable à un instant $t+1$ est définie par une équation Booléenne qui est fonction des autres états à l'instant t, voir même les instant précédents [47]. Ce modèle est le cadre général mais il existe plusieurs variants regroupés dans 3 grandes familles [48], les réseaux aléatoires (Random Boolean Network), les réseaux asynchrones et les réseaux probabilistes.

Les réseaux Booléens aléatoires ont été les premiers étudiés [46] du fait de leur simplicité qui requière une faible puissance de calcul. Ce sont des réseaux synchrones, parfaitement déterministes où tous les états sont mis à jour simultanément [49]. Le qualificatif d'aléatoire vient de leur grande dimension qui lors de l'inférence contraint en pratique à tester aléatoirement différentes structures de réseaux.

Dans les réseaux asynchrones [50] les nœuds sont mis à jour aléatoirement en série. Ils ont été développés pour corriger l'aspect déterministe et parfaitement synchrone des réseaux Booléens aléatoires pour être plus représentatif des observations biologiques. En contrepartie les réseaux asynchrones demandent plus de puissance de calcul. Les particularités de ces différents modèles non-déterministe sont revues dans [51]. On notera qu'un même réseau peut avoir des propriétés différentes si on considère un comportement synchrone ou asynchrone. Par exemple les attracteurs cycliques disparaissent dès que l'on considère de l'asynchronisme.

Le premier réseau Booléen probabiliste a été décrit par [52]. Dans ce modèle c'est la structure même du réseau qui est aléatoire et change au cours du temps. Ce modèle répond au constat que pour un gène donné, plusieurs règles permettent de recouper les données expérimentales. Plutôt que de choisir une seule règle, le modèle considère les règles les plus probables pour un même gène selon une chaine de Markov [53, 54]. De fait, ces modèles sont plus robustes et sont un compromis entre les réseaux Booléens basés sur des règles totalement

déterministes, et les réseaux Bayésiens qui reproduisent les comportements stochastiques observés sans description mécaniste.

1.3.4.2 Inférences de réseaux Booléens

L'inférence des réseaux Booléens a commencé avec l'arrivé des puces à ADN. La première étape commune à tous ces algorithmes d'inférence est la binarisation des données imposées par le formalisme du modèle. Le premier algorithme REVEAL [55] considère des réseaux synchrones. Il utilise une première étape d'identification des régulateurs pour chaque gène en se basant sur une mesure de l'information mutuelle. C'est un exemple d'approche mixte qui combine plusieurs concepts pour l'inférence de RRG. La seconde étape consiste à comparer le niveau d'expression entre 2 temps pour étudier les variations. Une approche de type "force brute" teste toutes les règles possibles avec 3 régulateurs au maximum par gène pour reproduire les données. Cette technique est limitée par la combinatoire et est sensible au bruit. La famille des méthodes BOOL de Akutsu [56, 57] reprend le même principe mais est plus adaptée aux données bruitées. Ce type de méthode a été appliqué avec succès pour l'inférence du réseau impliqué dans le cycle cellulaire de la levure avec un nombre de gène limité [58].

Les réseaux booléens probabilistes ont aussi été utilisés pour l'inférence de RRG [52, 59, 60]. L'algorithme décrit dans [59] est particulièrement intéressant car il exploite la cinétique des données en recherchant l'information mutuelle entre un potentiel régulateur à un instant t et le gène régulé à $t+1$. De plus, il utilise une métrique MDL pour "Minimum Description Length" qui considère conjointement une distance de recoupement avec les données et une distance de complexité du modèle. La distance de recoupement des données se base sur une notion de probabilité conditionnelle proche des réseaux Bayésiens dynamiques. Cette approche combine donc les principes de la théorie de l'information, de la théorie Bayésienne et des réseaux Booléens.

1.3.4.3 Avantages et limites

Les modèles booléens présentent l'avantage d'être simples à implémenter et faciles à exécuter. De plus, leurs propriétés mécaniste et dynamique permettent de reproduire des comportements observés dans des RRG expérimentaux et de faire des prédictions sur les états

stationnaires mais pas sur les transitoires. En contrepartie, ils sont limités à l'inférence de RRG avec un faible nombre de gènes du fait d'une combinatoire importante et du nombre de paramètres important. Les réseaux booléens déterministes sont en pratique peu robustes aux données bruitées. La binarisation nécessaire des données est également problématique puisqu'une partie importante de l'information est délaissée. Enfin, les méthodes comme REVEAL [55] et BOOL [56, 57] ne semblent pas être meilleures que le pur hasard [61].

1.3.5 Les approches par EDO

1.3.5.1 Les modèles EDO

Les modèles par équations différentielles ordinaires (EDO) sont mécanistes par excellence puisqu'ils sont par définition continus temporellement et quantitativement comme les systèmes biologiques. Dans ce formalisme on retrouve les notions d'états et de structure des RRG présentées dans la partie **1.2.1.2**. Ainsi, le RRG est modélisé par une fonction qui intègre des signaux externes (vecteur noté U) pour modifier son état interne (vecteur noté X) et générer des sorties observables (vecteur noté Y). La fonction qui définit la dynamique de ce système est noté F, et celle qui définit ces observables G. La dynamique de ce système est décrite par une EDO :

$$\begin{cases} \dfrac{dX(t)}{dt} = F(X(t), U(t)) \\ Y(t) = G(X(t), U(t)) \end{cases} \tag{5}$$

L'utilisation d'une équation différentielle en fonction du temps montre l'importance de l'histoire perçue par le système pour définir son état à un instant T. En effet, pour connaître l'état d'un système à un instant T il faut procéder à une intégrale temporelle des stimuli et de son état.

$$X(T) = \int_0^T F(X(t), U(t))dt \tag{6}$$

Si on continue le parallèle avec la différenciation, on retrouve la notion de mémoire dans la vision historique. On remarque aussi que la dynamique du RRG (définie par les fonctions F et G) ne se limite pas au stimulus, mais prend en compte l'état interne du système. On

retrouve l'idée que 2 cellules de types différents, donc d'état différent, avec la même structure réagissent différemment au même stimulus.

Tout comme les modèles Booléens, les EDO reproduisent les comportement dynamiques (oscillations, multi-stationnarité, robustesse, ...) mais ils sont en plus capables de reproduire les variations temporelles des variables mesurées. La contrepartie de cette représentativité réside dans le nombre élevé de paramètres qui définissent ces modèles qui en pratique sont non-identifiables. Il existe un panel de modèles allant des plus simples totalement linéaires [24, 62, 63], aux plus complexes avec des fonctions non-linéaires [64] et stochastiques [65]. La complexité du modèle a des conséquences sur ses propriétés [66], de fait le choix du modèle doit être un compromis entre sa représentativité biologique et son utilisation.

1.3.5.2 Inférence des modèles EDO

On peut inférer les modèles EDO à partir d'expériences de perturbation avec des données en état stationnaire ou avec des cinétiques. Dans le premier cas, l'hypothèse de stationnarité permet de simplifier les équations différentielles (cf eq 5) en annulant les termes dérivés. De plus, comme le stimulus est explicitement représenté, si on connait les cibles de la perturbation (qui peut être une drogue, un K.O. ou une sur-expression), on peut déduire les coefficients de la matrice jacobienne du modèle linéaire par une méthode de régression linéaire multiple. Cette méthode a été appliquée et validée dans [24] sur un réseau connu chez E.Coli contenant 9 gènes pour lesquels 9 expériences de sur-expression ont été réalisées. La même méthode a été appliqué dans [62] sur un réseau à l'échelle génomique sur un jeu de 515 expériences de perturbations chez la levure, toujours avec des mesures à l'état stationnaire. Dans ce cas les cibles des perturbations n'étaient pas connues mais inférées par une approche statistique. Dans ces 2 études, les modèles générés ont permis de prédire les cibles directes d'une nouvelle drogue non-utilisée dans le jeu de données initial.

Dans le deuxième cas, le modèle est inféré à partir de données cinétiques suite à une perturbation [63, 64, 65]. Dans [63] un modèle linéaire est utilisé pour inférer le même RRG avec les même données que dans [24]. Tout d'abord les données sont approximées par des fonctions lisses puis discrétisées afin d'estimer les dérivés des états à chaque point de mesure. De plus, la dimension des données est réduite via une analyse en composantes principales pour

rendre le problème identifiable. L'inférence de la matrice jacobienne se fait alors par simple inversion matricielle étant données les estimées des dérivés et des états. Pour les approches utilisant des modèles non-linéaires [64, 65], l'inférence utilise des méthodes heuristiques pour l'estimation des paramètres comme les algorithmes génétiques [64] ou "Maximum Likelihood Estimation" dans [65] qui utilise aussi les connaissances issues de la littérature pour définir la liste des régulateurs potentiels. Dans les 2 cas l'objectifs est de minimiser l'écart entre les données simulées et expérimentales en utilisant par exemple la méthode des moindres carrés.

1.3.5.3 Avantages et limites

Comme les autres approches Booléennes mécanistes et dynamiques, les méthodes par EDO ont l'avantage de reproduire des comportements observés et d'être prédictives même d'un point de vue quantitatif. Cette propriété est due à la pertinence biologique de ces modèles qui représentent explicitement le stimulus et inclus des processus non-linéaires. Cependant, en pratique ils sont limités par le nombre important de paramètres ce qui les rend difficilement identifiables. C'est pourquoi les méthodes basées sur les EDO ont recours à des hypothèses arbitraires de rareté des interactions ou à des linéarisations. Enfin, ces modèles qui se veulent mécanistes sont fondamentalement déterministes même si un bruit de type Gaussien peut être rajouté, ce qui comme nous allons l'exposer dans la suite, ne permet pas de reproduire le comportement observé à l'échelle d'une cellule.

1.4 Inférence des RRG à partir de données en cellule unique

1.4.1 Les données d'expression en cellule unique

Les techniques de biologie moléculaire ont connu un essor remarquable au cours des quinze dernières années notamment dans le domaine des études à l'échelle de la cellule unique. On sait depuis longtemps observer des cellules individuellement en microscopie et même quantifier le niveau de quelques protéines à l'échelle de la cellule grâce à la cytométrie de flux. Cependant, il n'existait pas encore de technique haut-débit pour accéder à la mesure des ARN à l'échelle de la cellule. Depuis, plusieurs techniques ont été développées en cellule unique basées sur le RNASeq ou la qPCR [67, 68]. En parallèle, d'autres techniques d'épigénomique

et génomique ont été développées à l'échelle de la cellule [69] mais ces dernières ont été rarement exploitées pour l'inférence des RRG comme nous le verrons dans l'étude des méthodes d'inférence alternatives. Les données en cellule unique offrent de multiples avantages. On peut sortir de l'effet de moyenne et observer l'hétérogénéité d'une population, comme dans le sang. Cela donne également accès au contenu interne d'une cellule, ce qui permet de lever la contradiction des biologistes qui décrivent le comportement d'une cellule à partir de mesure sur une population [70]. Enfin, la puissance statistique est importante car ces techniques permettent de mesurer des centaines voire des milliers de cellules individuellement, ce qui représente autant de réalisations statistiques indépendantes. Ce dernier aspect est certainement la raison de l'engouement de la recherche sur l'inférence des RRG pour l'analyse de ces nouvelles données comme nous allons le voir.

Cependant les premiers résultats ont suscité de la surprise, voir des doutes sur la crédibilité des données. En effet, les mesures ont montré une grande hétérogénéité inter-cellulaire, ce qui pouvait être attendu mais pas à ce niveau, ainsi qu'un nombre important de mesures nulles, ce qui a amené la communauté à caractériser ces distributions d'ARN de "zero inflated" [71]. Ces observations ont été interprétées de 2 façons bien différentes, ce qui a créé 2 stratégies distinctes dans l'utilisation de ces données pour l'inférence des RRG. La première est de considérer cette variabilité comme principalement du bruit technique et de la désynchronisation inter-cellulaire. La deuxième, qui sera notre approche, est de considérer la stochasticité comme une réalité biologique et de l'intégrer dans l'inférence. Nous allons maintenant détailler ces 2 approches.

1.4.2 Adaptation des algorithmes d'inférence de RRG pour l'analyse des données en cellule unique

Pour une très large partie, les approches d'inférence de RRG à partir de données en population que nous venons d'exposer ont été adaptées pour l'analyse de transcriptome en cellule unique. On trouvera dans [72, 73] des revues détaillées et dans [74] un test comparatif de performance d'inférence. La logique d'adaptation des algorithmes repose sur l'idée que l'hétérogénéité observée n'est rien d'autre que du bruit qui masque un processus d'expression fondamentalement déterministe et continue, conformément à la vision classique du proces-

sus de différenciation. Le bruit peut avoir une origine technique, appelée "dropout" [75] qui explique le nombre important de zéros, ou être due à la désynchronisation temporelle entre cellules au cours de la différenciation. Le bruit de mesure est déjà présent dans les données en population et nombre d'algorithmes existants ont été développés pour y être robustes. La gestion des zéros n'est donc qu'un cas particulier de bruit. En parallèle, l'hypothèse de désynchronisation a amené à reconstruire des trajectoires de "pseudo-temps" [76] pour trier les cellules chronologiquement et ainsi faire apparaître un processus déterministe continu. Cet ordonnancement temporel est basé sur le principe que 2 cellules proches temporellement sont proches quantitativement au niveau moléculaire puisqu'elles suivent le même processus. Par conséquent, toutes les hypothèses biologiques sur lesquelles reposent les algorithmes d'inférence développés sur des données en population sont toujours valides. Cela permet, en théorie, une utilisation directe de ces outils sur ces données pour bénéficier de leur puissance statistique en termes de répétition et de distribution jointe. Ainsi, SingleCellNet [77] et BoolTraineR [78] sont des algorithmes basés sur des réseaux booléens qui intègrent une première étape de tri des cellules temporellement. Des modèles de réseaux booléen asynchrones ont aussi été appliqués sur ce principe avec succès [79]. Dans la famille des approches statistiques par information mutuelle ou inférence bayésienne, on peut citer SCOUP [80], SCIMITAR [81] ou AR1MA1-VBEM [82] qui utilisent aussi une première étape de tri temporelle des cellules. Les approches par EDO conservent toujours les mêmes modèles utilisés pour l'analyse des données en population comme dans SCODE et InferenceSnapshot. Ces algorithmes ne nécessitent pas de donnée en cinétique puisque le temps est supposé reconstruit par ordonnancement des cellules.

1.4.3 Utilisation de modèles stochastique en cellule unique pour l'inférence des RRG

Même s'il existe toujours un doute sur la qualité de la quantification des transcrits dans une cellule, il existe depuis longtemps des observations biologiques qui témoignent d'une stochasticité biologique. Nous avons déjà mentionné le cas de comportements non déterministes lors de la différenciation. Mais plus directement, au niveau moléculaire, les données de cytométrie de flux montraient déjà des distributions dispersées de protéines sur une population

de cellule. Plus récemment, en 2002 Elowitz [83] a montré chez *Escherichia coli* que l'expression génétique était intrinsèquement aléatoire. Cette observation a depuis été confirmé chez d'autre organismes dont l'Homme [84, 85] et ont de plus montré que le régime dynamique d'expression était une succession de "bursts" d'ARN. Ces observations de bursts suivis de longues périodes de relaxation sans ARN sur plusieurs heures est compatible avec les formes des distributions des ARN données par les mesures en cellules uniques. Le nombre important de zéros correspondrait à des mesures pendant les périodes de relaxation, et la forte variabilité inter-cellulaire viendrait du caractère aléatoire de ces bursts. Cependant, il ne faudrait pas croire que tout ne serait que pur hasard et anarchie dans la cellule. Il a été montré que la fréquence et la taille des bursts sont corrélées à l'activité du gène [86]. Ainsi un gène faiblement exprimé aura une probabilité plus faible de générer des bursts contrairement à un gène fortement exprimé. Dans cette vision probabiliste, la régulation de l'expression génétique doit donc se voir comme la modulation des probabilités de bursts.

Les mécanismes moléculaires responsables de ces bursts sont encore mal connus. Cependant des observations au niveau des ARN naissants [87] montrent que cette stochasticité est présente au moment de la transcription. Une des hypothèses les plus répandues est que l'activité du promoteur est la source principale de stochasticité, en raison du nombre très limité de promoteur par gène dans une cellule (généralement égale au nombre de copie du chromosome). Le random telegraph, ou modèle à 2 états, proposé en 2005 modélise le mécanisme d'expression génétique en le résumant à deux états, "on" et "off", entre lesquels le promoteur oscille aléatoirement avec une fréquence moyenne k_{on} et k_{off}, conduisant ainsi respectivement à l'activation (on) ou l'inhibition (off) du gène [88, 89, 85]. C'est le basculement aléatoire entre ces deux conditions qui génère les bursts. Leur fréquence et leur durée moyenne sont dépendantes des paramètres k_{on} et k_{off}.

Pour l'inférence des RRG, si on considère que la stochasticité observée dans les données correspond bien à une réalité biologique, il est important de l'intégrer car elle est porteuse d'information et joue certainement un rôle dans le comportement du RRG. Très peu d'approche ont suivi cette idée. A notre connaissance seul l'algorithme SINCERITIES [90] intègre dans son algorithme le modèle à 2 états. Cependant le modèle random telegraph n'est pas en soi un modèle de RRG car aucun couplage n'est défini. Une étape préliminaire serait donc

la définition d'un modèle de RRG stochastique que nous aborderons dans l'article 2 de cette thèse.

1.5 Stratégies alternatives pour l'inférence de RRG

Dans les parties précédentes nous avons voulu donner un aperçu des différentes stratégies pour l'inférence des RRG à partir de données à l'échelle de la population ou de la cellule unique. Nous aurions pu aussi détailler les approches par réseaux de neurones [91, 92, 93] qui font partie de la famille des approches statistiques. Elles sont basées sur un processus d'apprentissage et nécessitent une grande quantité de données. Cependant, dans la pratique, beaucoup d'algorithmes combinent les méthodes que nous avons vu et il est donc difficile de les classer dans une de ces catégories. Par exemple [59] est une approche booléenne qui procède à une étape préliminaire d'inférence Bayésienne pour réduire la combinatoire.

Une autre approche intéressante est l'intégration de données hétérogènes. Toutes les méthodes citées jusqu'à présent se basent sur l'analyse d'un seul type de données basée sur la quantification des ARN. Pourtant il existe d'autres mesures expérimentales informatives en génomique, épigénomique et protéomique. Nous possédons aujourd'hui pas moins de dix techniques différentes permettant d'étudier les régulations épigénétiques de cellules individuelles [69]. Parmi les plus connues, nous pouvons noter le ChiP-seq pour les interactions ADN-protéine, la DNase-seq et l'ATAC-seq en ce qui concerne la structure chromatidienne, et le HiC pour l'organisation tri-dimensionnelle des chromosomes [69]. En protéomique il existe des nouvelles méthodes, telles que le CITE-Seq, qui semblent prometteuses et permettent déjà aujourd'hui d'accéder à une grande partie du compartiment protéique d'une seule cellule [94]. L'idée est que ces données dites "omic" contiennent une part d'information sur la régulation de l'expression génétique. Mais si elles sont analysées séparément, ces données ne permettent pas d'avoir une vue interconnectée des mécanismes. C'est pourquoi il est nécessaire d'intégrer toutes ces données "multi-omic" dans une analyse intégrative. Les méthodes existantes sont basées sur des méthodes statistiques de type data mining ou Bayésienne. Pour plus d'information sur ces méthodes intégratives nous conseillons cette revue récente [95].

Une autre stratégie est l'adaptation de ces méthodes aux importantes capacités de calcul parallèle disponible à l'heure actuelle. L'idée est de diviser le problème pour pouvoir le

paralléliser et ainsi contourner en partie de la problématique de la combinatoire. Une stratégie consiste à former des sous-groupes de gènes avec l'hypothèse qu'ils appartiennent à un sous-ensemble relativement interconnecté du réseau. L'idée est donc d'inférer en parallèle les sous-GRN puis éventuellement de les connecter ultérieurement, comme cela est proposé dans [96]. Cette stratégie a également été appliquée avec les réseaux de neurones [97].

1.6 Performances et limites de l'inférence des RRG

Suite à la multiplication importante du nombre d'algorithmes d'inférence de RRG, beaucoup d'études de tests comparatifs ont été menées [98, 99, 74] et même un concours international "DREAM Challenge" a été dédié à l'inférence des RRG [100]. Tous ces tests se basent sur la reconstruction de RRG connus issus de modèles *in silico* ou de modèles *in-vitro* censés être validés expérimentalement. Dans certain cas, il a même des RRG synthétiques qui ont été implémentés par génie génétique dans des bactéries [62]. Les RRG inférés sont alors comparés aux RRG réels à l'aide de différents indicateurs. Les conclusions de ces études comparatives varient dans leur enthousiasme sur la performance des algorithmes, mais globalement on note qu'il n'y a pas d'algorithme qui se démarque sensiblement des autres et que la capacité d'inférence est légèrement supérieure au pur hasard. Cette critique est sévère et décevante, d'autant que les méthodes basées sur l'analyse des données en cellule unique ne font pas beaucoup mieux que leur prédécesseur [74]. Ces résultats ne signifient pas que ces approches sont inutiles à la recherche comme le témoignent des succès [101] qui font avancer la connaissance scientifique. Mais l'objectif de reverse-engineering est encore loin d'être atteint.

De plus, il faut faire attention au crédit donné à ces études comparatives. Comme le soulignent eux-mêmes les organisateurs du challenge DREAM dans leur introduction [102], la question de l'évaluation des méthodes d'inférence est une problématique en soit. Les modèles *in silico* utilisés sont très certainement loin d'être représentatif de la réalité biologique, d'autant plus qu'ils n'intègrent pas le régime en burst de l'expression génétique. La définition de nouveaux modèles *in silico* standards basés sur le modèle du random telegraph serait une avancée pour la génération de données en cellule unique. Pour les RRG biologiques soi-disant connus et validés expérimentalement, il y a une contradiction fondamentale. S'il existait une

méthode expérimentale d'inférence des RRG, même très couteuse et longue, la problématique de l'inférence serait réglée ce qui n'est évidemment pas le cas. Ces études ne font que se baser sur des interprétations non formelles d'expériences qui font consensus dans la communauté scientifique. Concernant les RRG synthétiques, on ne peut ignorer les interactions avec les RRG endogènes qui vont certainement les perturber.

On pourrait aussi aborder la question des métriques d'évaluation couramment utilisées comme les ROC (Receiver Operating Characteristic) qui se concentrent sur le nombre d'interactions identifiées, mais pas sur la topologie globale du RRG. Nous reviendrons sur ce point juste après. La seule méthode acceptable de validation d'un algorithme d'inférence de RRG est la validation expérimentale répétée de prédictions générées par les RRG inférés. Un modèle est comme une théorie en physique, on ne peut pas démontrer qu'elle est vraie, on peut juste tester ses prédictions pour tenter de la rejeter. Cette méthode itérative est justement un des principes majeurs de la biologie dite des systèmes que nous souhaitons appliquer à moyen terme. Nous reviendrons sur ce point en fin de manuscrit dans l'examen des perspectives.

Bien que l'analyse critique des algorithmes existants ne soit pas simple comme on vient de le montrer, on peut néanmoins résumer ici une liste de leurs principales limites que nous avons abordées dans cette introduction :

— L'utilisation des corrélations pour l'inférence des interactions est problématique et il ne faut pas les considérer comme des causalités. Les corrélations ne peuvent que reproduire ce qui a été préalablement observé. Par conséquent la génération de prédiction par les RRG en réponse à un nouveau stimulus ou une modification de sa structure est impossible.

— La production de prédictions par simulation ne peut se faire que si la topologie du réseau est explicitement définie. Des algorithmes comme [81, 82, 103, 90] proposent des interactions associées à des indices de confiance indépendants généralement sous la forme d'une matrice d'interaction. La combinatoire de topologie de réseaux obtenus à partir de ces matrices est beaucoup trop importante pour pouvoir tous les simuler.

— Très souvent les protéines régulatrices considérées dans les RRG se limitent à des facteurs de transcription comme dans [78, 80, 81, 82, 103, 104]. Les interactions indirectes,

comme illustrées dans la figure 2, sont complètement ignorées alors qu'elles jouent un rôle aussi essentiel que les facteurs de transcriptions.

— La plupart des algorithmes n'utilisent qu'un seul type de données, en l'occurrence des mesures d'ARNm, en faisant l'hypothèse que le niveau des protéines est corrélé. Il existe de nombreux exemples de régulation post-traductionnelle qui infirment cette approximation comme [105] sur l'horloge circadienne.

— Le choix d'hypothèses biologiques trop simplificatrices est aussi une limite fondamentale des performances de l'inférence des RRG. Elles sont souvent justifiées par l'utilisation d'outils statistiques puissants capables d'analyser des données de milliers de gènes sur des milliers de cellules, mais le prix à payer en représentativité biologique est certainement trop élevé.

Dans cette thèse nous proposons de surmonter un certain nombre des limites de l'inférence des RRG en adoptant une approche mécaniste comme je l'ai pratiqué dans mon expérience antérieure en ingénierie. Pour cela il est nécessaire d'étudier les propriétés du système cellulaire à son échelle unitaire (article 1). Il faudra ensuite développer un modèle mécaniste de RRG réaliste biologiquement (article 2) sur lequel reposera une nouvelle stratégie d'inférence (article 3) qui utilisera au maximum les capacités disponibles de calcul dont nous disposons (article 4).

2 Résultats

2.1 Article 1 : Single-Cell-Based Analysis Highlights a Surge in Cell-to-Cell Molecular Variability Preceding Irreversible Commitment in a Differentiation Process

Dans un premier temps nous avons caractérisé l'évolution de l'expression au niveau de la cellule sur notre modèle biologique de différenciation. Comme nous l'avons vu dans l'introduction bibliographique, différents modèles théoriques suggèrent que le processus de différenciation s'accompagne d'une augmentation de la variabilité de l'expression génique [21, 106, 107].

Notre équipe a analysé expérimentalement la variabilité de l'expression génique au cours du processus de différenciation érythrocytaire, grâce à des technologies récentes permettant d'accéder au contenu moléculaire de plusieurs cellules, individuellement. Cette analyse a fait l'objet d'une publication dans le journal PLoS Biology (article 1) [108].

2.1.1 Principaux résultats de l'article 1

Dans un premier temps, l'expression de 110 gènes, sélectionnés sur la base d'une analyse RNASeq préalablement réalisée dans l'équipe, a été mesurée par RTqPCR dans des populations de progéniteurs érythrocytaires dénommés T2EC [13], en état d'auto-renouvellement (0h), ou induites en différenciation depuis 8h, 24h, 48h et 72h. A l'échelle de la population, la variabilité entre les populations de cellules est expliquée par le stade de la différenciation. Cependant, les analyses en cellules uniques sur un sous ensemble de 92 gènes mesurés sur 96 cellules aux mêmes stades de différenciation montrent que la variabilité inter-cellulaire n'est que faiblement expliquée par la différenciation. Il y a donc une autre source de stochasticité interne très importante.

Afin de tester l'hypothèse selon laquelle l'engagement en différenciation s'accompagne d'une forte augmentation de la variabilité de l'expression génique, nous avons utilisé la mesure de l'entropie. Ainsi, pour chaque temps de différenciation, nous avons calculé une valeur d'entropie par gène et comparé la distribution de ces valeurs au cours de la différenciation (figure 8 de l'article 1). Nos résultats montrent que l'entropie, augmente significativement

à 8h, reste stable jusqu'à 24h, et diminue progressivement jusqu'à 72h, suggérant ainsi que la différenciation érythrocytaire s'accompagne d'un pic de la variabilité d'expression génique intercellulaire à 8h-24h du processus.

Nous avons ensuite cherché une cause à cette augmentation de la variabilité. Nous avons exclu une variation du cycle cellulaire, cependant une variation du volume a été constaté à 48h soit 40h après l'augmentation de l'entropie, ce qui suggère que la variation du volume est plutôt une conséquence de la variation de l'entropie. Enfin, j'ai démontré en utilisant le modèle 2 états de l'expression génétique que l'asynchronie entre les cellules au cours de la différenciation ne peut pas être la cause de l'augmentation transitoire de la variabilité car les simulations montrent au contraire une diminution du pic d'entropie lorsque l'on considère l'asynchronie (figure 9E et 9F de l'article 1).

La mesure du pic de variabilité entre 8h et 24h suggère l'idée d'un engagement de la cellule. Afin de tester cette hypothèse, les T2EC ont été induites à se différencier pendant 24h ou 48h, puis remises dans le milieu d'auto-renouvellement. Les résultats de cette expérience montrent qu'après 24h de différenciation, les cellules sont encore capables de s'auto-renouveler, alors qu'après 48h, elles ne prolifèrent plus (figure 10A de l'article 1). Le point de non-retour de l'engagement des T2EC dans le processus de différenciation semble donc se situer entre 24h et 48h.

Ce comportement dynamique m'a alors amené à proposer l'hypothèse selon laquelle l'information du stimulus de différenciation se propage dans le réseau avec une certaine inertie. Afin de valider cette hypothèse j'ai comparé les distributions marginales de chaque gène entre les différents temps qui montrait de façon significative que certains gènes étaient régulés entre 0h et 8h, puis un autre groupe entre 8h et 24h et ainsi de suite. Ces résultats n'ont pas été publiés, mais afin d'identifier le premier groupe de gènes régulés, dénommés "early genes", une cinétique sur la réponse initiale a été spécialement réalisée pour mesurer l'expression des gènes en cellules uniques à 0h, 2h, 4h et 8h après différenciation. Elle a permis d'identifier des "early genes" activés entre 0h et 2h puis entre 2h et 4h (figure 7 de l'article 1). Parmi ces gènes, on retrouvait notamment un cluster impliqué dans la synthèse des stérols qui avait déjà été observé sur les cinétiques en population (figure S4 de l'article 1).

2.1.2 Principales conclusions de l'article 1

Ces travaux montrent très clairement qu'à l'échelle de la cellule la stochasticité est très importante d'un point de vue quantitatif et qu'elle évolue dynamiquement avec un pic transitoire qui précède l'engagement de la cellule. En contrepartie, l'effet de la différenciation sur l'expression est indéniable au niveau de la population. Ces résultats montrent bien la pertinence de l'utilisation d'un modèle stochastique de RRG pour analyser les données d'expression en cellule unique pour d'une part reproduire cette variabilité individuelle, et d'autre part contraindre cette stochasticité par les interactions pour guider la différenciation au niveau de la population. Le modèle 2 états utilisé pour les simulations *in silico* montre bien sa pertinence. Toutefois ce modèle ne considère pas d'interaction entre gènes, ce qui sera réalisé dans le cadre de l'article 2.

L'observation de "vagues d'expression" dans le GRN a permis de valider expérimentalement le concept de vague que je détaille dans l'article 3 et qui sera le principe central de ma stratégie d'inférence des RRG.

2.1.3 Article 1

Publié le 27 Décembre 2016 dans le journal PLoS Biology

URL : https ://doi.org/10.1371/journal.pbio.1002585

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

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

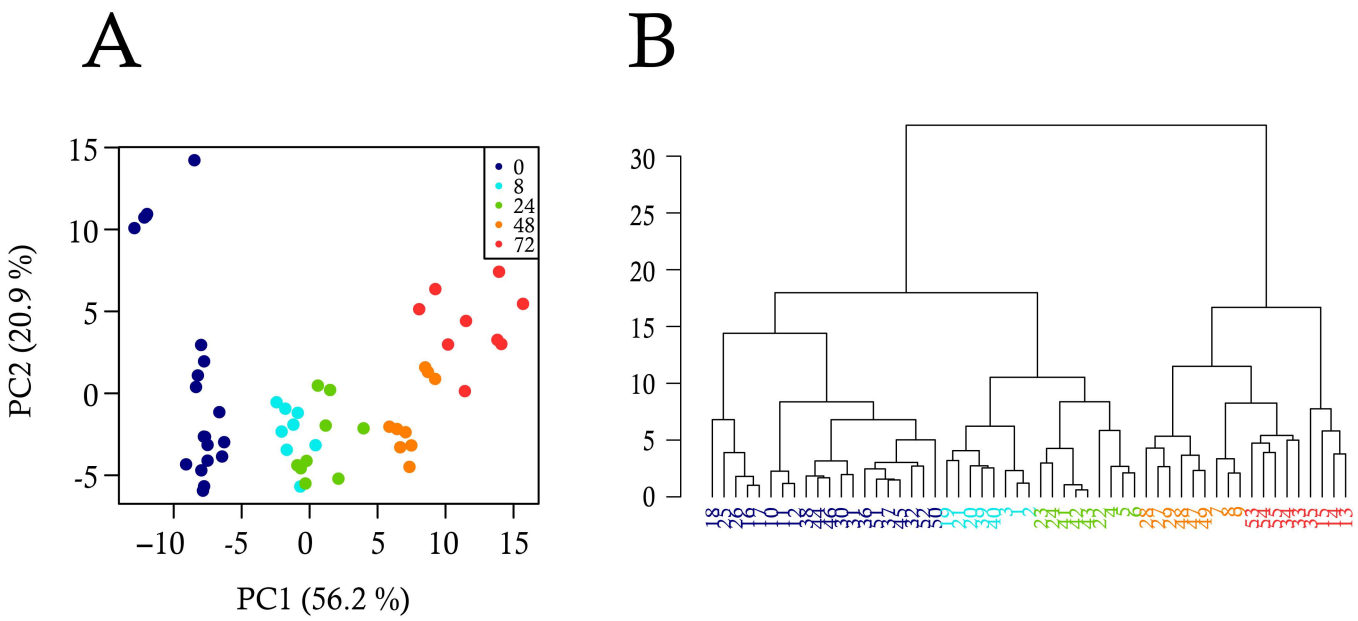


Fig 1. Analysis of bulk-cell gene expression during the differentiation process. Gene expression data were produced by RT-qPCR in triplicate from three independent T2EC populations collected at five differentiation time-points (0 h, 8 h, 24 h, 48 h, 72 h). The expression level of 110 genes (18 invariants, 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

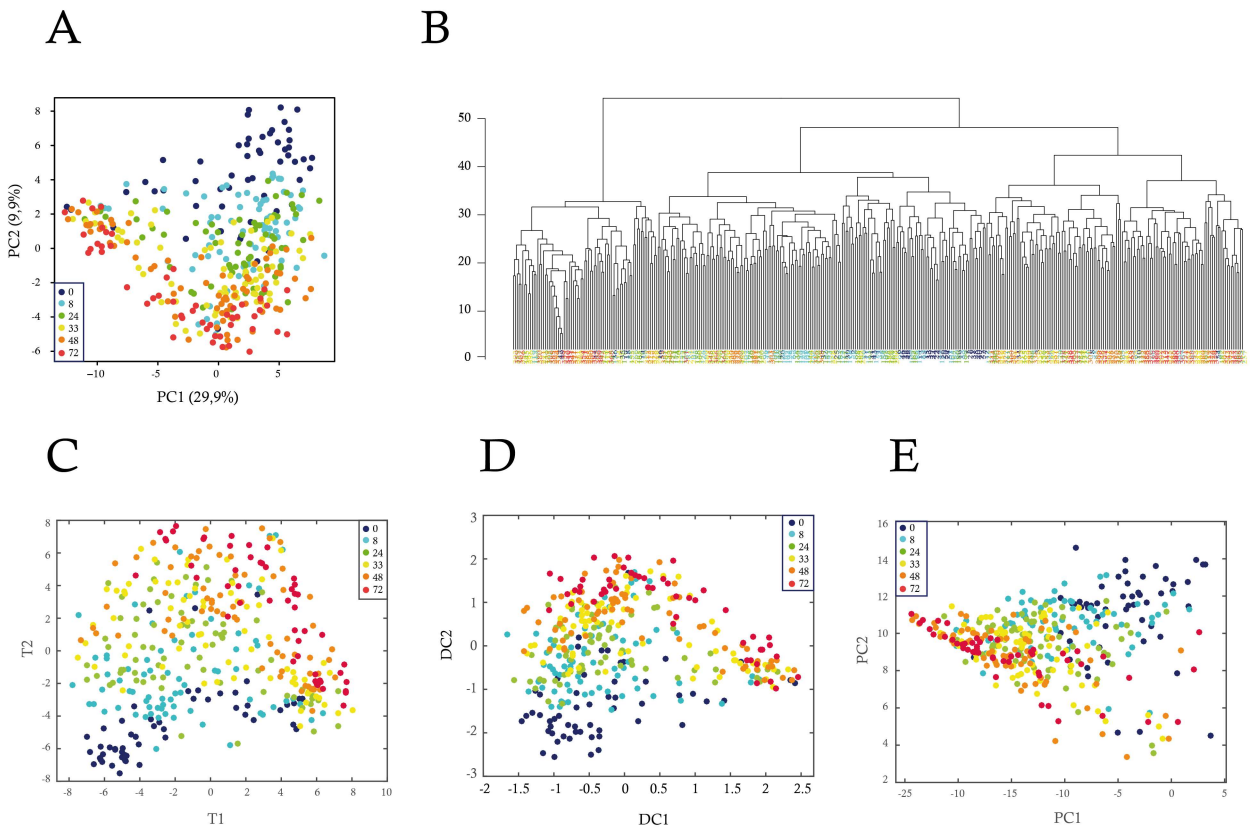


Fig 2. Analysis of single-cell gene expression during the differentiation process. Gene expression data were produced by RT-qPCR from individual T2EC collected at six differentiation time-points (0, 8, 24, 33, 48, and 72 h). The expression of 90 genes was analyzed in single-cells by five different multivariate 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.

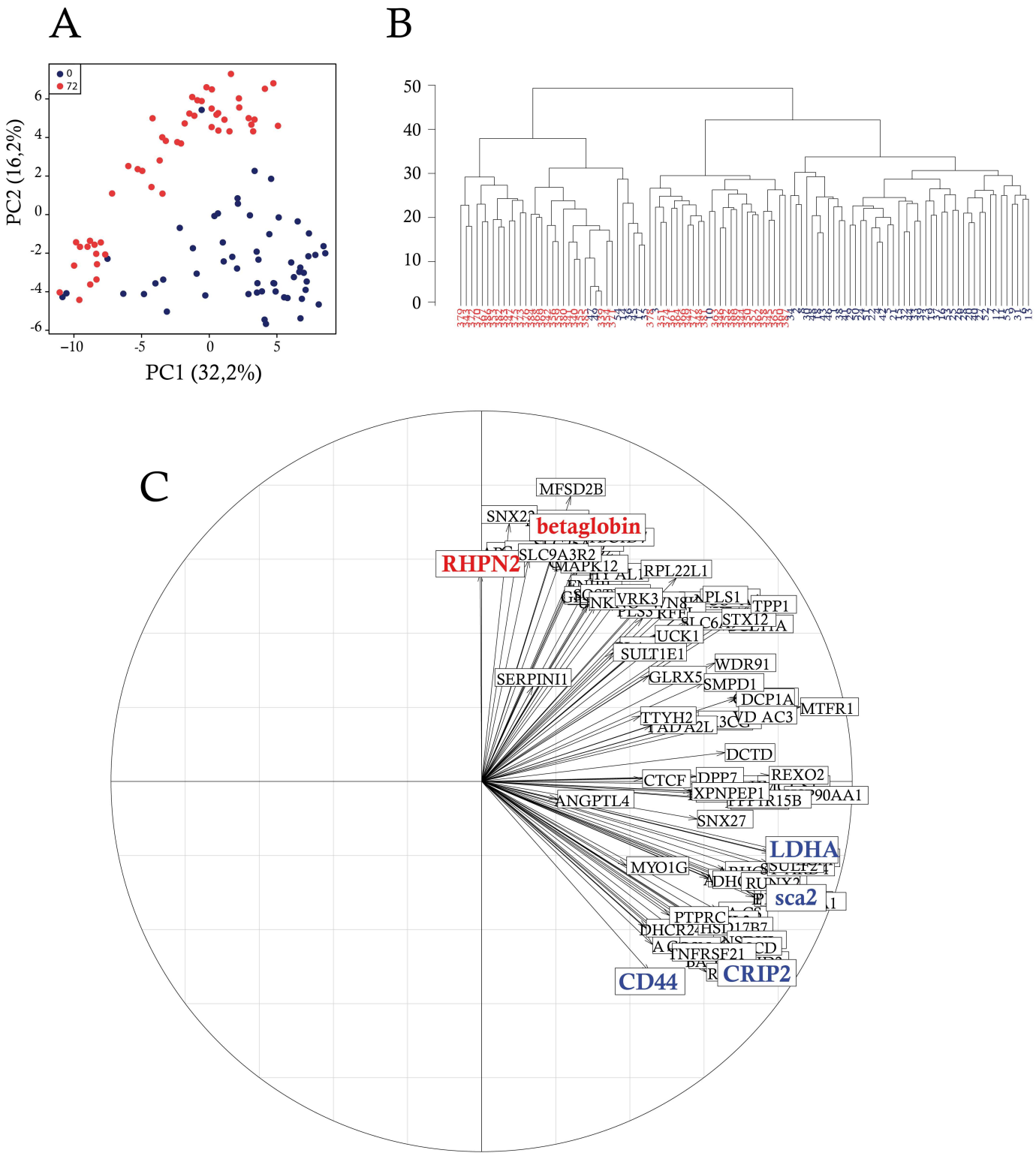


Fig 3. Gene expression-based discrimination between self-renewing and differentiating individual cells. Single-cell gene expression data were analyzed considering only self-renewing cells and cells induced to differentiate since 72 h. (A) Principal component analysis (PCA); (B) Hierarchical cluster analysis (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

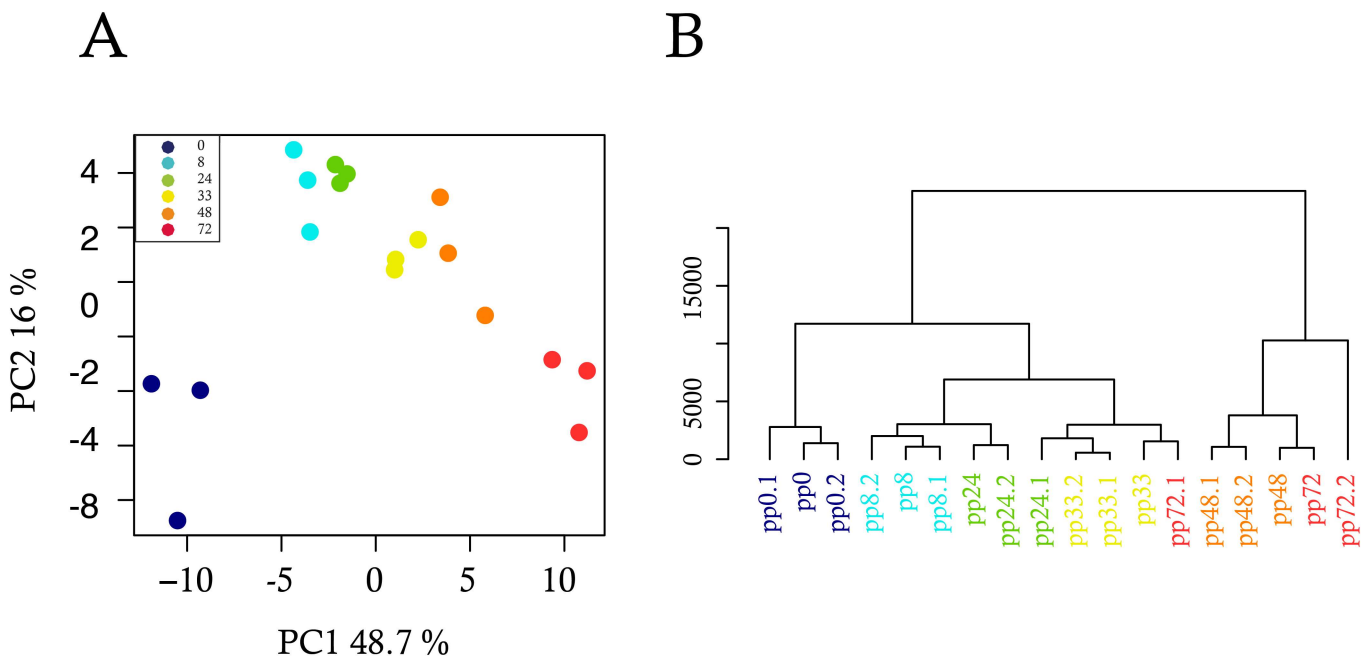


Fig 4. Analysis of single-cell data averaged over pseudo-populations. We separated single-cells into three pseudo-populations with around one-third of single cells for each time-point. We then calculated the average gene expression over each pseudo-population, and analyzed the resulting averaged data 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).

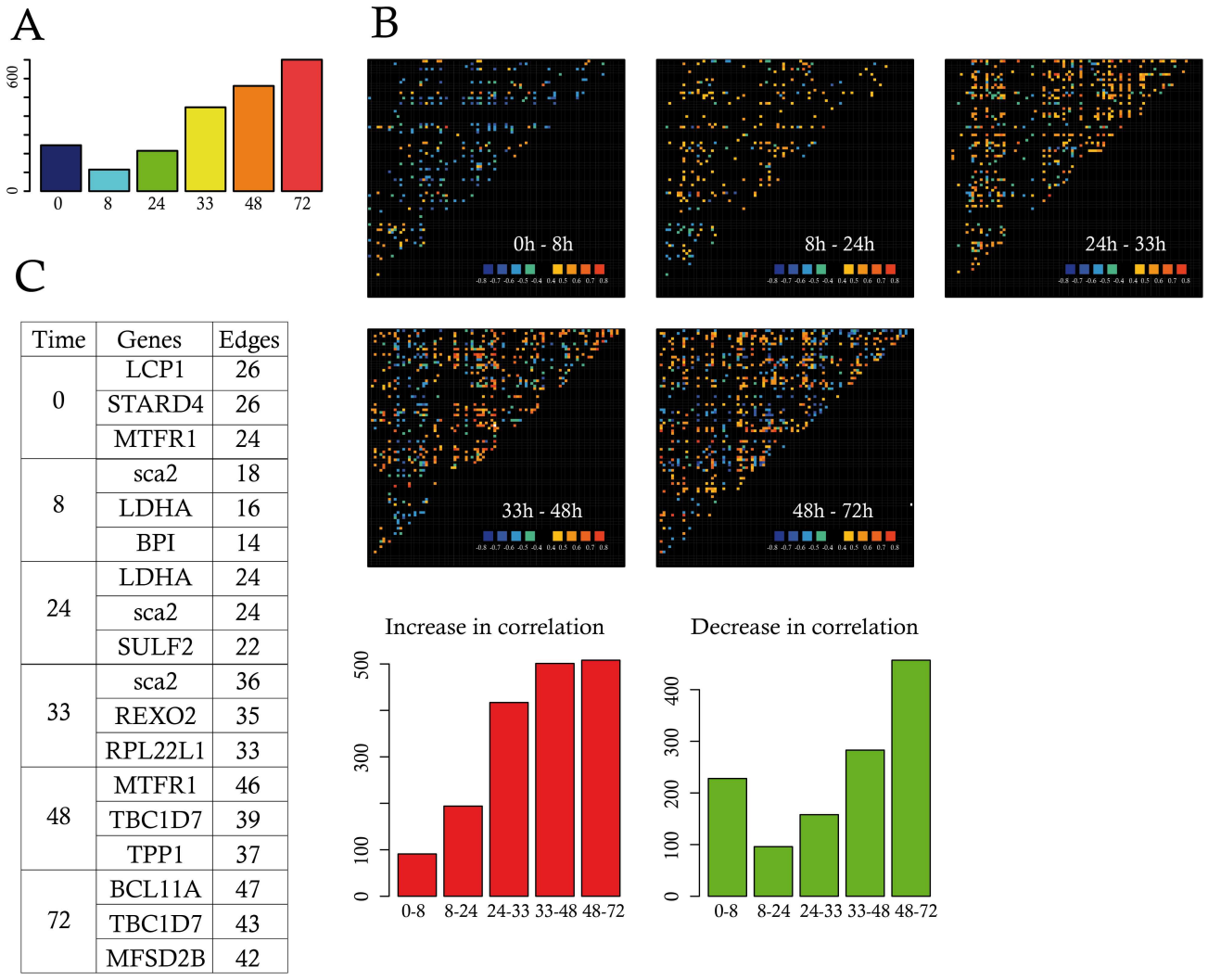
The PCA result of the pseudo-population therefore looked much more like the population than the single-cell results. Similarly, HCA generated a clustering that was not quite as clear as the analysis of bulk RNA data, but much better than the single-cell analysis (Fig 4B). The HCA results showed for example 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 sub-sampled 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 sub-sampling 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.

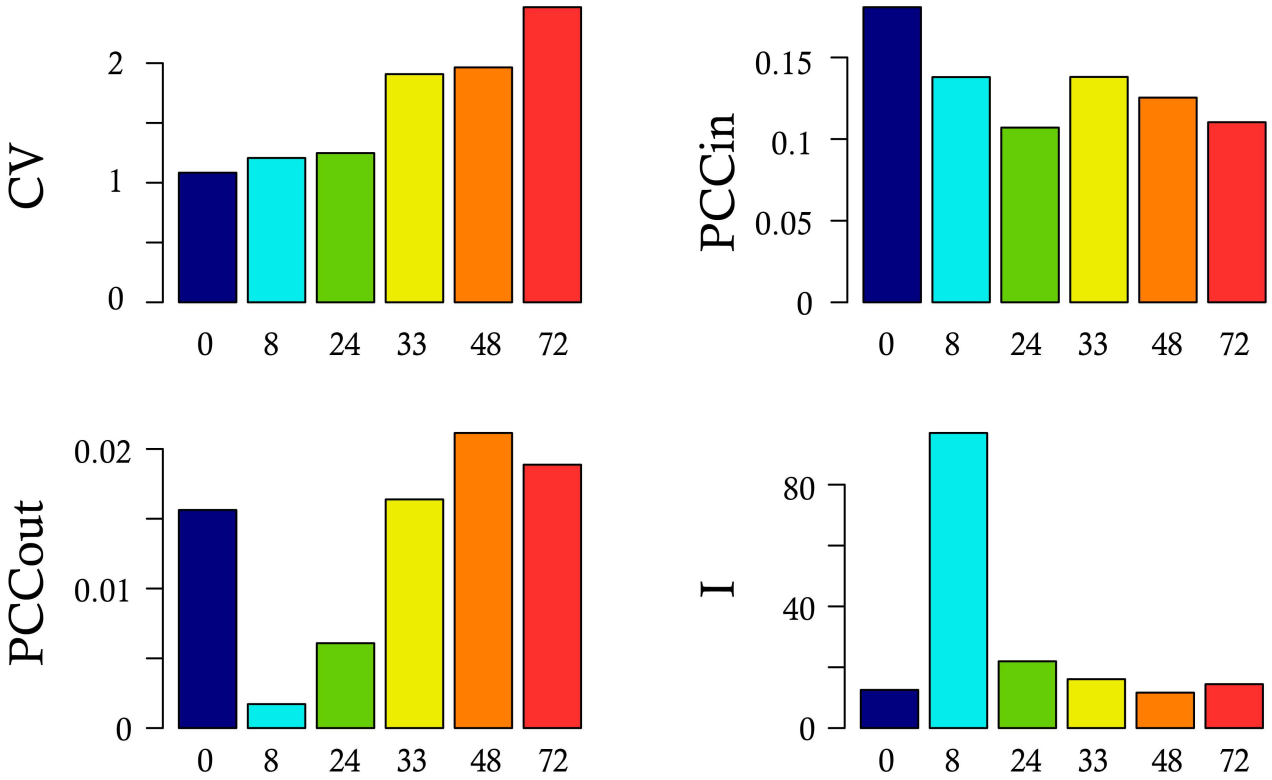


Fig 6. Identification of a dynamical network biomarker. Shown is the behavior of a subset composed of 12 genes fitting the following criteria: increase in their standard deviation and participation to increasing correlations, between 0h and 8h. For this subset, we plotted the mean coefficent of variation (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

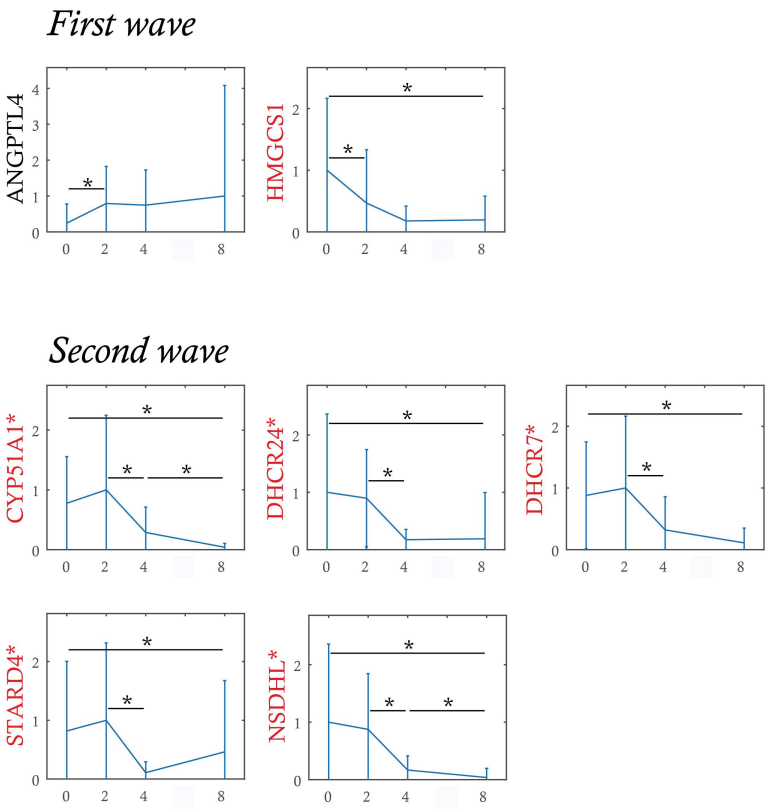


Fig 7. Initial expression waves analysis. Genes are sorted according to the time of the first significant expression variation. The first wave corresponds to genes with a significant variation detected during 0 h and 2 h. The second wave corresponds to genes with a significant variation detected during 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.

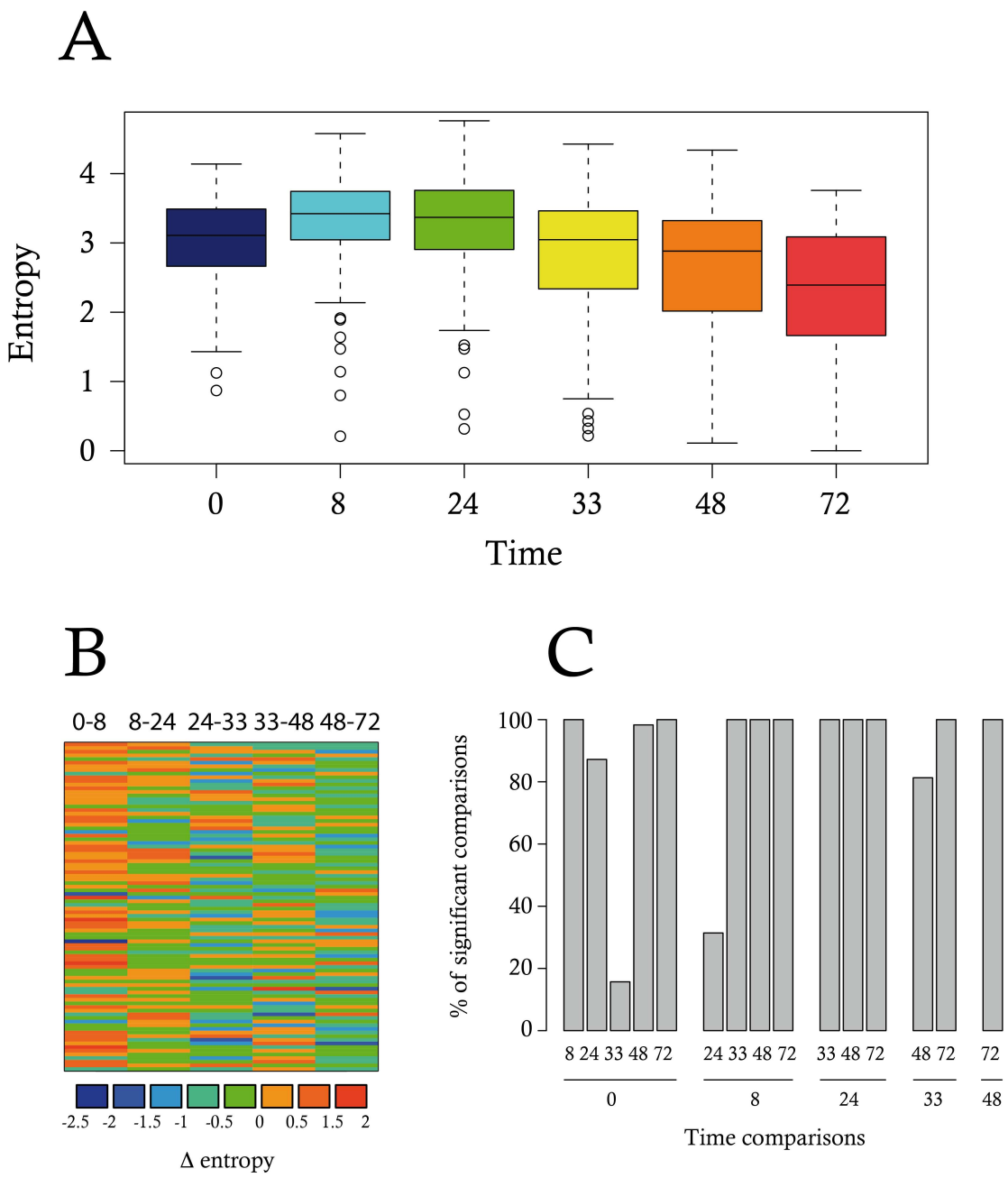


Fig 8. Cell-to-cell heterogeneity measurement using Shannon entropy. (A) A Shannon entropy was calculated for each time-point for each gene. Boxplots represent the distribution of the entropy values; (B) Gene entropy variation: for each gene (i.e., lines), we represented the difference between entropy 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 reordering 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.

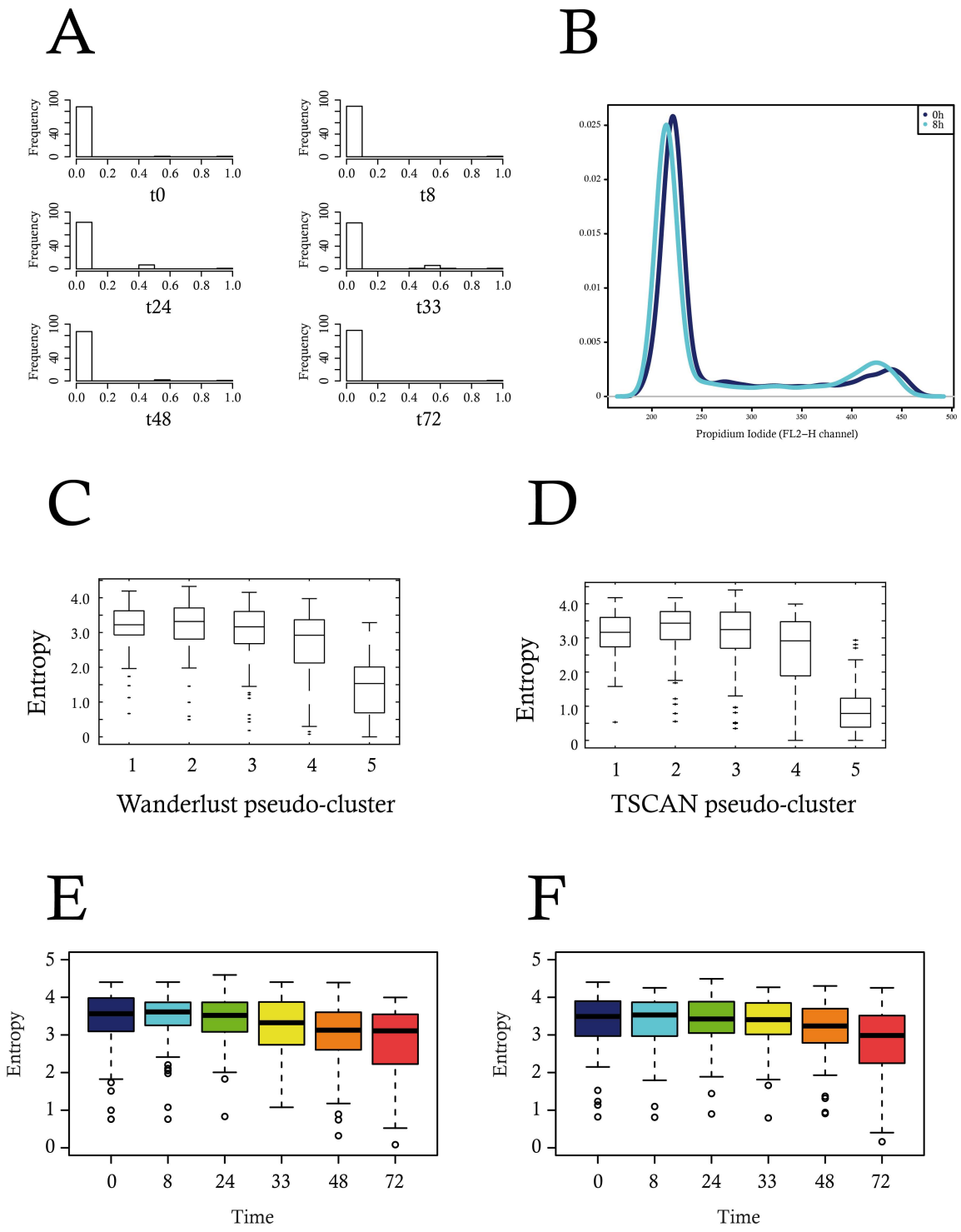


Fig 9. Exploration of potential cofounding factors. (A) Correlation of the CTCF gene with the rest of the 91 genes, at all six time-points. (B) FACS analysis of the cell cycle repartition at 0 and 8 h of differentiation. The difference between the two distributions was found not to be statistically significant ($p = 0.18$ 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

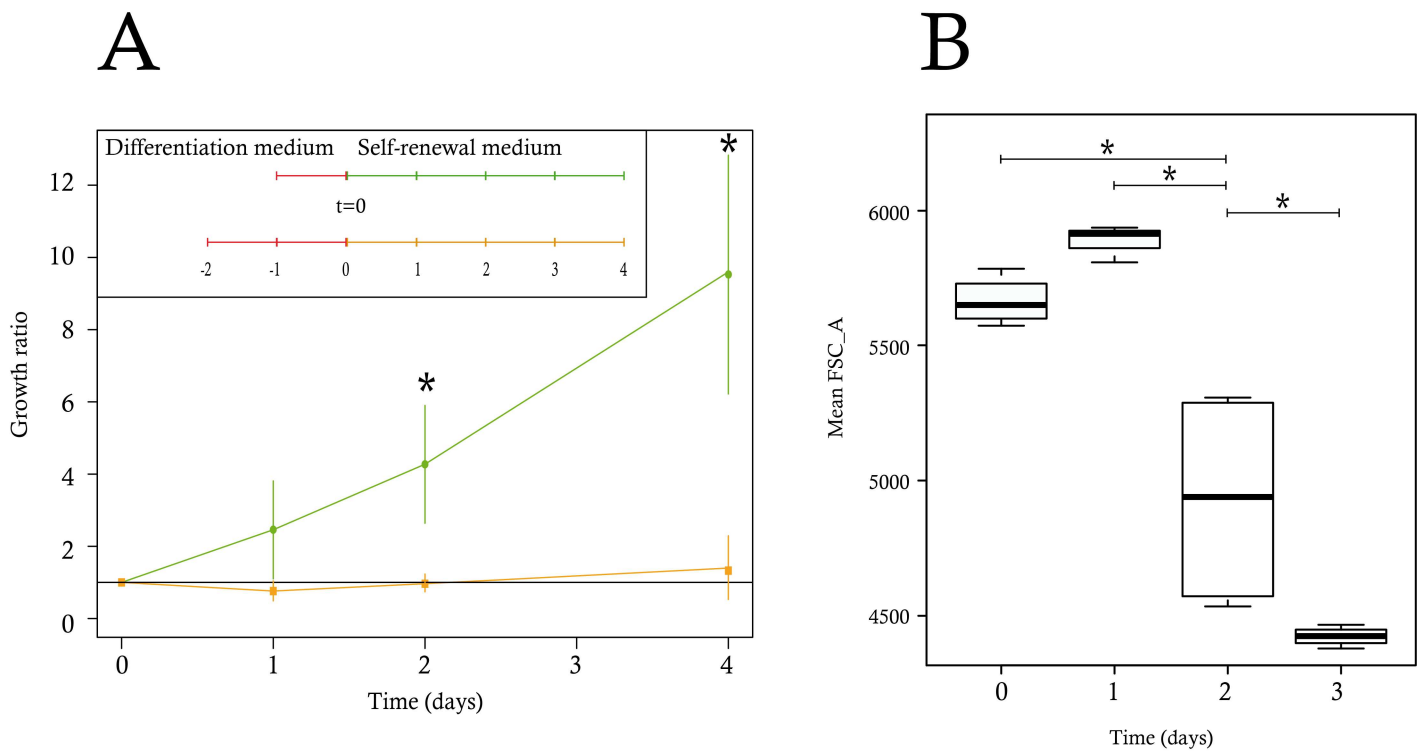


Fig 10. Evolution of physiological differentiation parameters. (A) T2EC were induced to differentiate for 24 and 48 h and subsequently seeded back in self-renewal conditions. Cells were then counted every day for 5 d. The green curve represents the growth of cells induced to differentiate for 24 h and 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 SuperScript 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

2.2 Article 2 : Inferring gene regulatory networks from single-cell data : a mechanistic approach

Comme nous l'avons détaillé dans l'introduction, nous avons choisi d'adopter une approche mécaniste pour inférer les relations de causalité dans les RRG. Dans cet article nous définissons un modèle stochastique mécaniste de RRG qui sera utilisé dans une approche mathématique du problème d'inférence.

2.2.1 Principaux résultats de l'article 2

A partir du modèle d'expression stochastique à 2 états présenté en introduction, nous avons dérivé un modèle hybride plus simple (équation 4 de l'article 2) dénommé "Piecewise Deterministic Markov Process" (PDMP). Dans ce modèle, seul le promoteur est défini par un processus stochastique. Les ARN et protéines sont des variables continues dont l'évolution est définie par des équations différentielles qui dépendent de l'état du promoteur. Puis, afin de modéliser un RRG avec des interactions causales, nous avons défini une fonction d'interaction (équation 10 de l'article 2) basée sur des hypothèses mécanistes qui exprime les fréquences de transition k_{on} et k_{off} du gène régulé en fonction des concentrations des protéines des gènes régulateurs. C'est ce modèle dit de PDMP couplés qui sera utilisé dans l'article 3.

Dans l'optique d'une approche mathématique de la problématique d'inférence, nous avons dérivé un modèle statistique capable de donner une approximation de la distribution jointe des ARN en régime stationnaire (équation 12 de l'article 2). Cette approximation reproduit bien les distributions stationnaires du modèle PDMP couplés dans le cas d'un RRG de type "toggle switch" (figure 6 de l'article 2). Ce modèle statistique recoupe également bien (figure 7 de l'article 2) la distribution marginale stationnaire *in vitro* d'un gène mesurée dans le cadre de l'article 1.

Le modèle statistique a ensuite été utilisé dans une approche de maximisation de la vraisemblance (équation 13 de l'article 2) pour inférer des petits RRG à 2 gènes. Les données expérimentales *in silico* ont été générées à partir du modèle de PDMP couplés. 10 petits RRG ont été tous inférés avec succès (figure 8 de l'article 2).

2.2.2 Principales conclusions de l'article 2

Nous avons obtenu un modèle stochastique mécaniste de RRG sous la forme de PDMP couplés qui est un bon compromis entre représentativité biologique et simplicité. En ajoutant une hypothèse de stationnarité il est en pratique possible d'inférer des petits RRG en dérivant un modèle statistique.

2.2.3 Article 2

Publié le 21 Novembre 2017 dans le journal BMC Systems Biology.

URL : https ://doi.org/10.1186/s12918-017-0487-0

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

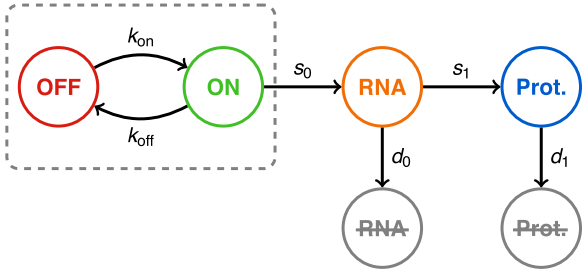


Fig. 1 Scheme of the two-state model of gene expression. We use it as the basic block of our network model

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

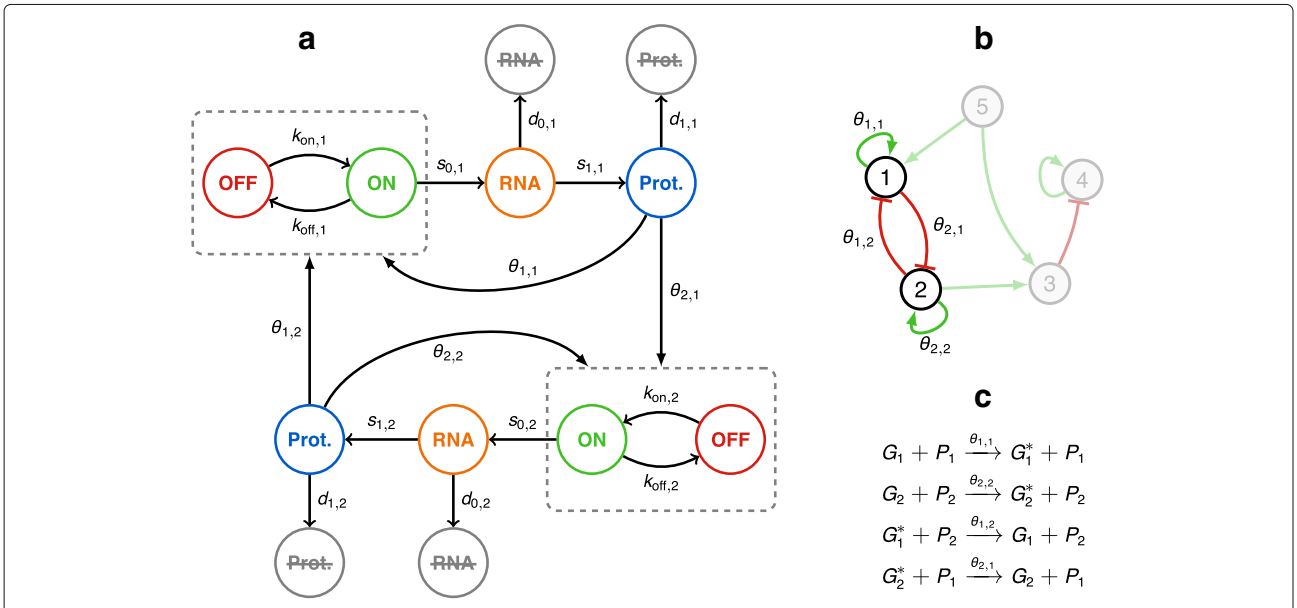


Fig. 2 Different views of the network model. **a** Two genes interacting with each other, forming a network. Interactions are assumed to arise from the dependence of promoter dynamics on protein quantities. **b** A higher level of abstraction leads to the traditional gene regulatory network representation. **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. $(k_{\text{on},i} +$

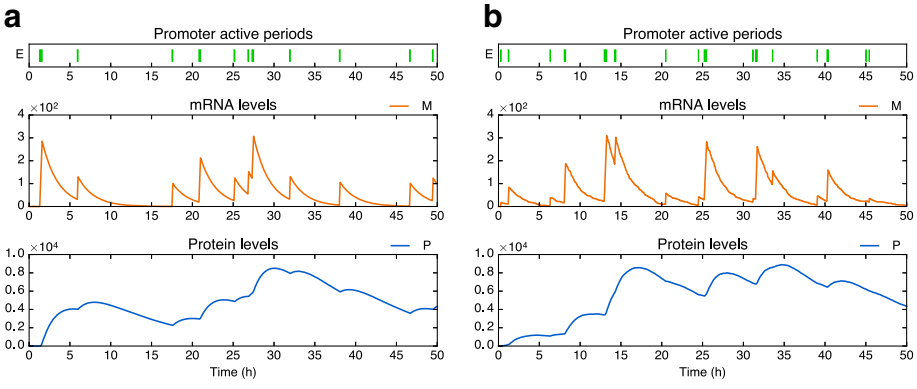


Fig. 3 Simulations of the two-state model for a single gene. **a** Sample path of the PDMP model using our hybrid numerical scheme (computation time ≈ 0.05 s). **b** Sample path of the classic model using exact stochastic simulation [27] (computation time ≈ 10 s). Parameters values are $k_{\text{on}} = 0.34$, $k_{\text{off}} = 10$, $s_0 = 10^3$, $s_1 = 10$, $d_0 = 0.5$ 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

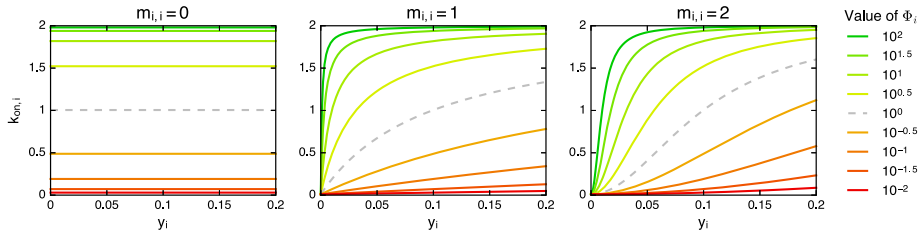


Fig. 4 Different auto-activation types in the network model. Each color corresponds to a fixed value of Φ_i in formula (10), and each curve represents $k_{\text{on},i}$ as a function of y_i for $m_{i,i} = 0$ (no feedback), $m_{i,i} = 1$ (monomer-type feedback) and $m_{i,i} = 2$ (dimer-type feedback). 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

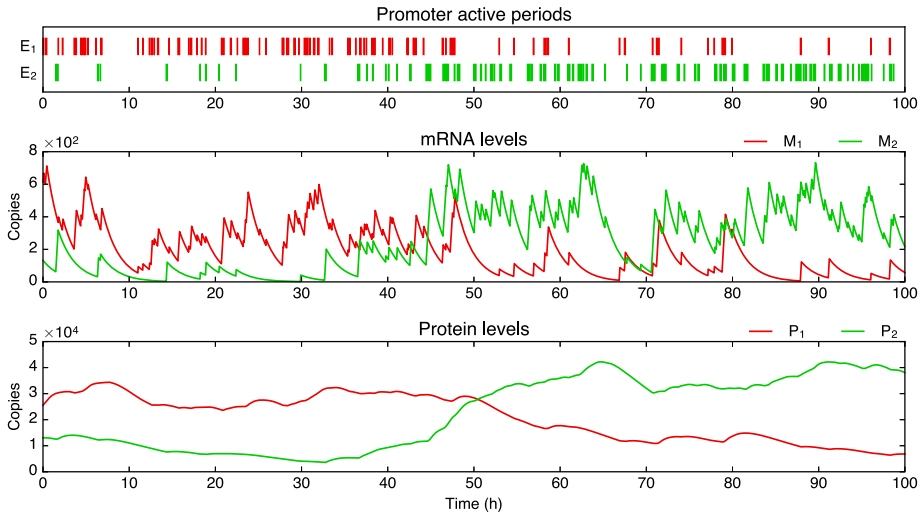


Fig. 5 Sample path of a two-gene toggle switch. The first gene is plotted in red and the second in green. While always staying in a bursty regime regarding mRNAs, genes can switch between high and low frequency modes (here at $t \approx 50$ h). From this example, it is clear that the overall joint distribution can contain correlations even if the bursts themselves are not coordinated

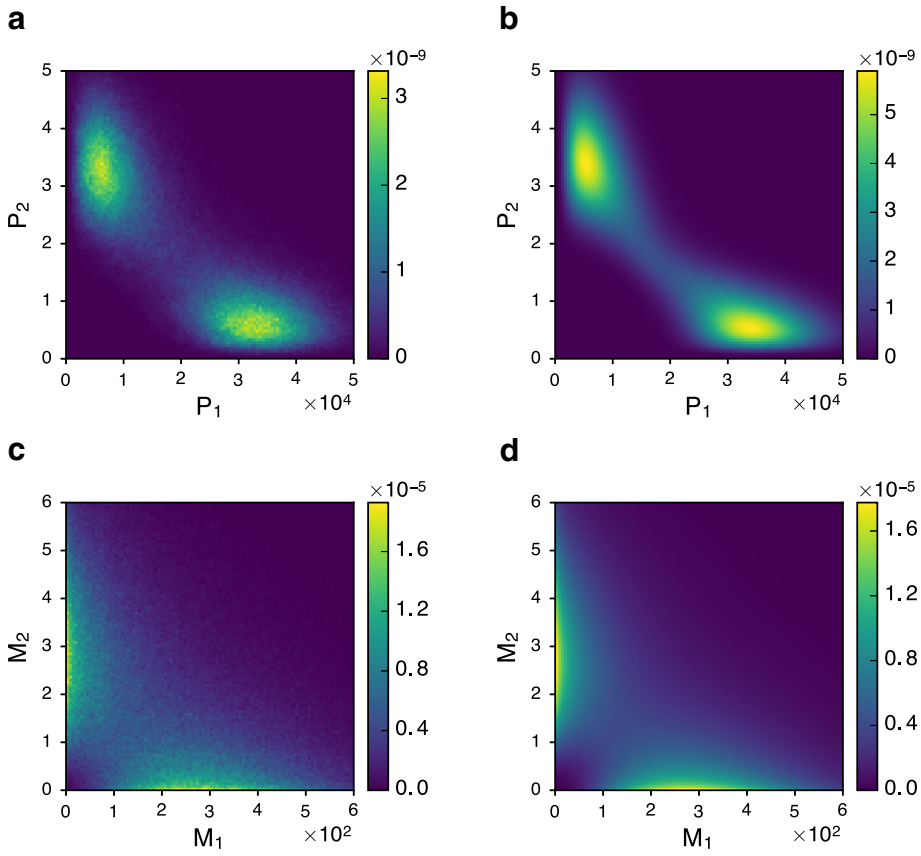


Fig. 6 Exact and approximate stationary distributions for the example of toggle switch. True distributions (left side) were estimated by sample path simulation, while approximations (right side) have explicit formulas. **a** True distribution of proteins. **b** Approximate distribution of proteins, from 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 $a = k_{\text{on}}/d_0$ and $b = k_{\text{off}}/d_0$), most of the mass of the distribution $\text{Beta}(a, b)$ is located at $x \ll 1$ so we have the first order approximation

$$\begin{aligned} x^{a-1}(1-x)^{b-1} &= x^{a-1}\exp((b-1)\log(1-x)) \\ &\approx x^{a-1}\exp(-bx) \end{aligned}$$

and thus $\text{Beta}(a, b) \approx \gamma(a, b)$. 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].

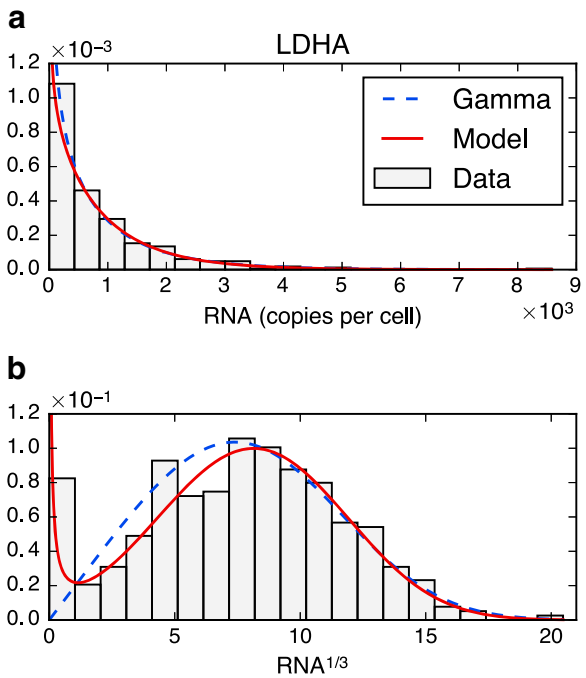


Fig. 7 Fitting marginal distributions from real single-cell data: example of the LDHA gene. The red curve is the stationary distribution associated with our interaction form (here a single gene with auto-activation), while the dashed blue curve corresponds to the basic two-state model in the bursty regime (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

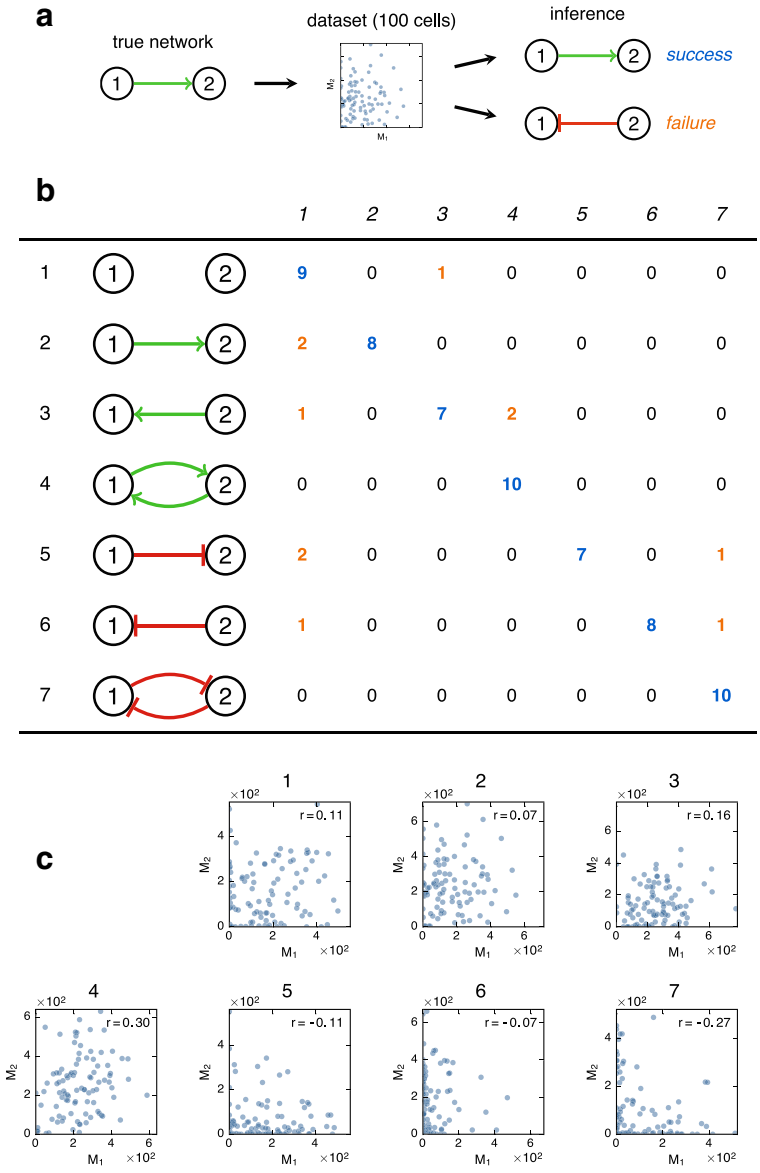


Fig. 8 Testing our inference method on simple networks. **a** For each network, numbered from 1 to 7, we simulated 100 cells using the full mechanistic model until the stationary regime was reached. Then we took a snapshot of their mRNA levels and inferred the parameters from this data. The result was 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.

Inferring gene regulatory networks from single-cell data: a mechanistic approach

– Supplementary information –

Ulysse Herbach[1,2,3*], Arnaud Bonnaffoux[1,2,4], Thibault Espinasse[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, 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] The CoSMo company, 5 passage du Vercors, 69007 Lyon, France

Contents

[*] ulysse.herbach@ens-lyon.fr

1 First simplifications

1.1 Normalizing the PDMP network model

In this section we detail the normalization of our network model. Recall that the original model is defined by

$$\begin{cases} E_i(t) : 0 \xrightarrow{k_{\mathrm{on},i}} 1, \;\; 1 \xrightarrow{k_{\mathrm{off},i}} 0 \\ M_i{}'(t) = s_{0,i}E_i(t) - d_{0,i}M_i(t) \\ P_i{}'(t) = s_{1,i}M_i(t) - d_{1,i}P_i(t) \end{cases} \tag{1}$$

where $k_{\mathrm{on},i} = k_{\mathrm{on},i}(P_1, \ldots, P_n)$ and $k_{\mathrm{off},i} = k_{\mathrm{off},i}(P_1, \ldots, P_n)$. First we observe that, given an initial condition

$$(E_1^0, \ldots, E_n^0) \in \{0,1\}^n, \quad (M_1^0, \ldots, M_n^0) \in \prod_{i=1}^n \left[0, \frac{s_{0,i}}{d_{0,i}}\right], \quad (P_1^0, \ldots, P_n^0) \in \prod_{i=1}^n \left[0, \frac{s_{0,i}s_{1,i}}{d_{0,i}d_{1,i}}\right],$$

the system stays in this set for all $t > 0$, and we introduce the dimensionless variables:

$$\overline{M}_i = \frac{d_{0,i}}{s_{0,i}} M_i \in [0,1] \quad \text{and} \quad \overline{P}_i = \frac{d_{0,i}d_{1,i}}{s_{0,i}s_{1,i}} P_i \in [0,1].$$

Then, since $s_{0,i}$, $s_{1,i}$, $d_{0,i}$ and $d_{1,i}$ are constants, we get

$$\overline{M}_i{}'(t) = \frac{d_{0,i}}{s_{0,i}} M_i{}'(t) = d_{0,i}\left(E_i(t) - \frac{d_{0,i}}{s_{0,i}} M_i(t)\right) = d_{0,i}\left(E_i(t) - \overline{M}_i(t)\right)$$

and

$$\overline{P}_i{}'(t) = \frac{d_{0,i}d_{1,i}}{s_{0,i}s_{1,i}} P_i{}'(t) = d_{1,i}\left(\frac{d_{0,i}}{s_{0,i}} M_i(t) - \frac{d_{0,i}d_{1,i}}{s_{0,i}s_{1,i}} P_i(t)\right) = d_{1,i}\left(\overline{M}_i(t) - \overline{P}_i(t)\right).$$

As a result, we obtain the normalized model:

$$\begin{cases} E_i(t) : 0 \xrightarrow{\overline{k_{\mathrm{on},i}}} 1, \;\; 1 \xrightarrow{\overline{k_{\mathrm{off},i}}} 0 \\ \overline{M}_i{}'(t) = d_{0,i}\left(E_i(t) - \overline{M}_i(t)\right) \\ \overline{P}_i{}'(t) = d_{1,i}\left(\overline{M}_i(t) - \overline{P}_i(t)\right) \end{cases} \tag{2}$$

where the rescaled interaction function $\overline{k_{\mathrm{on},i}}$ is defined by

$$\overline{k_{\mathrm{on},i}}(\overline{P}_1, \ldots, \overline{P}_n) = k_{\mathrm{on},i}\left(\frac{s_{0,1}s_{1,1}}{d_{0,1}d_{1,1}}\overline{P}_1, \ldots, \frac{s_{0,n}s_{1,n}}{d_{0,n}d_{1,n}}\overline{P}_n\right)$$

and $\overline{k_{\mathrm{off},i}}$ is defined analogously. It is straightforward to see that, given a path

$$(E_i(t), \overline{M}_i(t), \overline{P}_i(t))_i$$

of the normalized model (2), the corresponding path of the original model (1) is

$$\left(E_i(t), \frac{s_{0,1}}{d_{0,1}}\overline{M}_i(t), \frac{s_{0,1}s_{1,1}}{d_{0,1}d_{1,1}}\overline{P}_i(t)\right)_i.$$

In this sense, both models are equivalent: in the main text and in the next sections, we always consider model (2) but forget the "bars" to keep the notations simple.

1.2 Separating mRNA and protein timescales

Here we justify the reduced network model involving only promoters and proteins, which is valid when $d_{1,i} \ll d_{0,i}$ for all gene i. A full proof is beyond the scope of this article but we provide a heuristic explanation. We temporarily drop the i index for simplicity. Let $t_1 \geqslant t_0 \geqslant 0$ and $E \in \{0,1\}$, and suppose $E(t) = E$ for all $t \in [t_0, t_1]$. Moreover, let $M_0 = M(t_0) \in [0,1]$ and $P_0 = P(t_0) \in [0,1]$. If $d_1 < d_0$, the solution of the linear ODE system

$$\begin{cases} M' = d_0(E - M) \\ P' = d_1(M - P) \end{cases}$$

is given for $t \in [t_0, t_1]$ by

$$\begin{cases} M(t) = E + (M_0 - E)e^{-d_0(t-t_0)} \\ P(t) = E + (P_0 - E)e^{-d_1(t-t_0)} + \dfrac{d_1}{d_0 - d_1}(M_0 - E)\left(e^{-d_1(t-t_0)} - e^{-d_0(t-t_0)}\right) \end{cases}$$

Hence, if $d_1 \ll d_0$, we have

$$P(t) \approx E + (P_0 - E)e^{-d_1(t-t_0)}$$

using the fact that $|M_0 - E| \leqslant 1$ and $|e^{-d_1(t-t_0)} - e^{-d_0(t-t_0)}| \leqslant 1$, and thus $P(t)$ approximates the solution of the differential equation $P' = d_1\,(E - P)$.

2 Hartree approximtaion

2.1 Hartree approximation for the PDMP model

Before deriving the approximation, we introduce some notation. Let n be the number of genes in the network, $\mathcal{E} = \{0,1\}^n$ and $\Omega = (0,1)^n$. At time t, promoter and protein configurations are denoted by $E_t = (e_1, \ldots, e_n) = e \in \mathcal{E}$ and $P_t = (y_1, \ldots, y_n) = y \in \Omega$, respectively. The distribution of (E_t, P_t) then evolves along time according to its Kolmogorov forward (aka master) equation, which is a linear partial differential equation (PDE) system in our case. This system is high dimensional ($|\mathcal{E}| = 2^n$, the number of possible promoter configurations) but the associated linear operator contains lots of zeros. Using the tensor product notation $\otimes$, one can write down the equation in a compact form:

$$\frac{\partial u}{\partial t} + \sum_{i=1}^{n} \frac{\partial\,(F_i u)}{\partial y_i} = \sum_{i=1}^{n} K_i u \tag{3}$$

where $u(t,y) = (u_e(t,y))_{e\in\mathcal{E}} \in \mathbb{R}^{2^n} \simeq (\mathbb{R}^2)^{\otimes n}$ represents the probability density function (pdf) of (E_t, P_t), and matrices $F_i(y_i)$, $K_i(y) \in \mathcal{M}_{2^n}(\mathbb{R}) \simeq \mathcal{M}_2(\mathbb{R})^{\otimes n}$ are defined by

$$F_i(y_i) = I_2 \otimes \cdots \otimes \underbrace{F^{(i)}(y_i)}_{i} \otimes \cdots \otimes I_2, \qquad K_i(y) = I_2 \otimes \cdots \otimes \underbrace{K^{(i)}(y)}_{i} \otimes \cdots \otimes I_2$$

with

$$F^{(i)}(y_i) = \begin{pmatrix} -d_{1,i}y_i & 0 \\ 0 & d_{1,i}(1 - y_i) \end{pmatrix} \quad \text{et} \quad K^{(i)}(y) = \begin{pmatrix} -k_{\text{on},i}(y) & k_{\text{off},i}(y) \\ k_{\text{on},i}(y) & -k_{\text{off},i}(y) \end{pmatrix}.$$

The sum in the left side of equation (3) clearly corresponds to a deterministic transport term, while the right side corresponds to the stochastic transitions between promoter configurations.

Furthermore, the PDE system comes with the boundary condition

$$\forall i \in \{1, \dots, n\}, \quad F_i u = 0 \quad \text{on } \partial\Omega \tag{4}$$

and the probability condition

$$u \geqslant 0 \quad \text{and} \quad \forall t \in \mathbb{R}_+, \quad \sum_{e \in \mathcal{E}} \int_\Omega u_e(t, y) \, \mathrm{d}y = 1. \tag{5}$$

The self-consistent "Hartree" approximation consists in splitting this 2^n-dimensional problem into n independent 2-dimensional problems by "freezing" the y_j for $j \neq i$ where i is fixed, and then gathering the solutions by taking their tensor product to produce an approximation of the true pdf (see [1] for a heuristic explanation in the discrete protein setting). More precisely, one reduced problem is derived for each gene i from (3)-(4)-(5):

$$\frac{\partial u^i}{\partial t} + \frac{\partial (F^{(i)} u^i)}{\partial y_i} = K^{(i)} u^i \tag{6}$$

where $u^i(t, y) = (u_0^i(t, y), u_1^i(t, y))^\top \in \mathbb{R}_+^2$ satisfies the initial condition $u^i(0, y) = u^{i,0}(y)$, the boundary condition $F^{(i)}(y_i) u^i(y) \to 0$ when $y_i \to 0$ or 1, and the probability condition $\int_0^1 [u_0^i(t, y) + u_1^i(t, y)] \, \mathrm{d}y_i = 1$ for all $t \geqslant 0$ and $y_1, \dots, y_{i-1}, y_{i+1}, \dots, y_n \in (0, 1)$. Therefore, each u^i is a pdf with respect to $(e_i, y_i) \in \{0, 1\} \times (0, 1)$ but not on $\mathcal{E} \times \Omega$. Finally, the Hartree approximation is given by

$$u(t, y) \approx \bigotimes_{i=1}^{n} u^i(t, y) \tag{7}$$

where the equality holds if for all i, $k_{\mathrm{on},i}$ and $k_{\mathrm{off},i}$ only depend on y_i.

2.2 Solving the reduced problem

For the moment, the time-dependent closed-form solution of (6) is unavailable, but the unique stationary solution can be easily obtained if one knows a primitive of

$$\lambda_i : y_i \mapsto \frac{k_{\mathrm{on},i}(y)}{d_{1,i} y_i} - \frac{k_{\mathrm{off},i}(y)}{d_{1,i}(1 - y_i)}$$

which is the nonzero eigenvalue of the matrix $M^{(i)} = K^{(i)} (F^{(i)})^{-1}$. Indeed, letting $v^i = F^{(i)} u^i$, the stationary equation for v^i from (6) becomes

$$\frac{\partial v^i}{\partial y_i} = M^{(i)} v^i$$

and then, crucially using the fact that $M^{(i)}$ has a constant eigenvector $(-1, 1)^\top$ associated with eigenvalue λ_i (the other eigenvalue being 0), one can check that $v^i = e^{\varphi_i} (-1, 1)^\top$ is a solution when $\frac{\partial \varphi_i}{\partial y_i} = \lambda_i$. If one has such a φ_i, the stationary solution of (6) is given by

$$u_0^i(y) = {Z_i}^{-1} y_i^{-1} \exp(\varphi_i(y)) \quad \text{and} \quad u_1^i(y) = {Z_i}^{-1} (1 - y_i)^{-1} \exp(\varphi_i(y)) \tag{8}$$

where Z_i is the normalizing constant (which may still depend on y_j for $j \neq i$). Note that the existence of a positive constant α such that $\min(k_{\mathrm{on},i}, k_{\mathrm{off},i}) \geqslant \alpha$ imposes the limit 0 for $\exp(\varphi_i(y))$ when $y_i \to 0$ or 1, and thus the boundary condition is satisfied. We also obtain the promoter probabilities $p_{0,i} = p(e_i = 0) = Z_{0,i}/Z_i$ and $p_{1,i} = p(e_i = 1) = Z_{1,i}/Z_i$ where $Z_{0,i} = \int_0^1 y_i^{-1} \exp(\varphi_i(y)) \, \mathrm{d}y_i$, $Z_{1,i} = \int_0^1 (1 - y_i)^{-1} \exp(\varphi_i(y)) \, \mathrm{d}y_i$ and $Z_i = Z_{0,i} + Z_{1,i}$.

In particular, when $k_{\mathrm{on},i}$ and $k_{\mathrm{off},i}$ do not depend on y_i (i.e. no self-interaction), we get

$$\varphi_i(y) = \frac{k_{\mathrm{on},i}(y)}{d_{1,i}} \log(y_i) + \frac{k_{\mathrm{off},i}(y)}{d_{1,i}} \log(1-y_i)$$

which gives the classical solution

$$u_0^i(y) = \frac{b_i}{a_i+b_i} \cdot \frac{y_i^{a_i-1}(1-y_i)^{b_i}}{\mathrm{B}(a_i, b_i+1)} \quad \text{and} \quad u_1^i(y) = \frac{a_i}{a_i+b_i} \cdot \frac{y_i^{a_i}(1-y_i)^{b_i-1}}{\mathrm{B}(a_i+1, b_i)} \tag{9}$$

with $a_i = k_{\mathrm{on},i}(y)/d_{1,i}$ and $b_i = k_{\mathrm{off},i}(y)/d_{1,i}$. This form makes clear the promoter probabilities $p_{0,i}$ and $p_{1,i}$ and the conditional distributions of protein y_i given the promoter state $e_i = 0$ or 1, both being Beta distributions. Since the state is usually not observed, one usually considers the marginal pdf of y_i, which is also a Beta:

$$\underline{u}^i(y) = u_0^i(y) + u_1^i(y) = \frac{y_i^{a_i-1}(1-y_i)^{b_i-1}}{\mathrm{B}(a_i, b_i)}. \tag{10}$$

Note that the conditional distribution of mRNA given proteins also has the form (10) since the PDMP equation is the same, although the argument is not the Hartree approximation but rather the more common quasi-steady state assumption.

2.3 Protein marginal distribution

Given the form of the solution (8), it is in fact always straightforward to integrate over promoters, even for the full (stationary) Hartree approximation (7), and we finally obtain

$$\underline{u}(y) = \sum_{e\in\mathcal{E}} u_e(y) \approx \sum_{e\in\mathcal{E}} \left[\bigotimes_{i=1}^n u^i(y)\right]_e = \sum_{e\in\mathcal{E}} \prod_{i=1}^n \frac{\exp(\varphi_i(y))}{Z_i(y)|e_i - y_i|} = \prod_{i=1}^n \frac{\exp(\varphi_i(y))}{Z_i(y) y_i(1-y_i)} \tag{11}$$

where we recalled the possible dependence of Z_i on some y_j. Hence, when φ_i and Z_i are known functions, one gets a fully explicit approximation of the joint protein distribution.

3 Explicit interactions

Here we derive an explicit form for the interactions between genes, starting from a coarse-grained biochemical model. That is, for a given gene i, we focus on defining functions $k_{\mathrm{on},i}(y_1, \ldots, y_n)$ and $k_{\mathrm{off},i}(y_1, \ldots, y_n)$ where $y_1, \ldots, y_n$ denote the protein quantities. For simplicity, we drop the i index in this section when there is no ambiguity.

3.1 Simple biochemical model

The basic idea is to slightly refine the two-state model of gene expression: in addition to the usual switching reactions (whose rates are k_{on} and k_{off}), 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. Finally, we assume that the chromatin transitions are due to fast interactions with ambiant proteins (binding, hit-and-run, etc.) so that the promoter-switching reactions always see chromatin in its quasi-stationary 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

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

where p_0 (resp. p_1) is the probability of the 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 $y_1, \dots, y_n$. We consider 2^n permissive configurations and 2^n non-permissive configurations as follows: for all $I \subset \mathcal{G}$ where $\mathcal{G} = \{1, \dots, n\}$, 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 $C_\emptyset$ (non-permissive) and $C_\emptyset^*$ (permissive), which describe dynamics when no protein is present, according to the reactions

$$C_\emptyset \xrightarrow{\alpha} C_\emptyset^*, \quad C_\emptyset^* \xrightarrow{\beta} C_\emptyset.$$

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

$$C_I^* + P_j \xrightarrow{a_j} C_{I\cup j}^* + P_j, \quad C_{I\cup j}^* \xrightarrow{b_j} C_I^*,$$

$$C_I + P_j \xrightarrow{c_j} C_{I\cup j} + P_j, \quad C_{I\cup j} \xrightarrow{d_j} C_I.$$

The system then evolves with $[C_I], [C_I^*] \in \{0, 1\}$ and $\sum_I [C_I] + [C_I^*] = 1$, 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 S1, writing $\overline{a}_j = a_j[P_j]$ and $\overline{c}_j = c_j[P_j]$ for simplicity. 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 – 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 for the chromatin when none of the P_j are present).

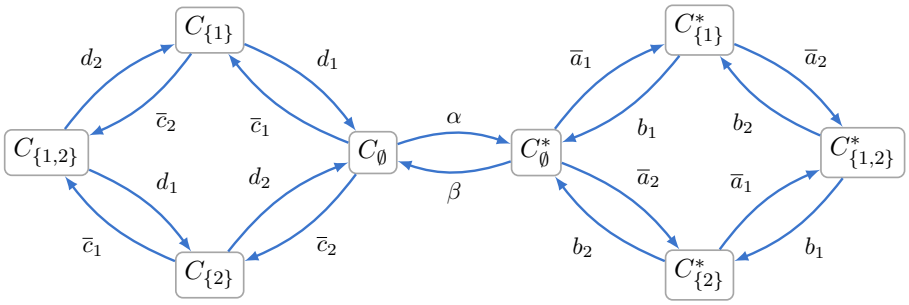


Figure S1: Chromatin states and transitions rates in the case of $n = 2$ proteins.

3.2 Stationary distribution

Letting $\mathcal{S} = \{0,1\}^{n+1}$, each state can be coded by a vector $s = (s_0, s_1, \ldots, s_n) \in \mathcal{S}$ 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. More precisely, the probability π_s of the chromatin being in state $s \in \mathcal{S}$ is given by

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

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,

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

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

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

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

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

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

$$k_{\text{on}} = \frac{k_0 \beta \prod_{j=1}^n (\mu_j [P_j] + 1) + k_1 \alpha \prod_{j=1}^n (\lambda_j [P_j] + 1)}{\beta \prod_{j=1}^n (\mu_j [P_j] + 1) + \alpha \prod_{j=1}^n (\lambda_j [P_j] + 1)}. \tag{12}$$

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 ($\lambda_j < \mu_j$).

3.3 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 simply replacing $[P_j]$ in equation (12) with a function of $[P_j]$ corresponding to the quasi-stationary concentration of the complex. This approximation seems to be relevant to capture the overall dependence of k_{on} on the proteins, the main point being to use a continuous description (e.g. rate equations) for proteins, which are abundant, while keeping a discrete (stochastic) description for chromatin. We chose to replace $[P_j]$ with $[P_j]^{m_j}$ where $m_j > 0$, 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.

3.4 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 (12) 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 (12). 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

$$k_{\text{on}} = \frac{k_0\beta \prod_{j\neq i}(\mu_j[P_j]^{m_j} + 1) + k_1\alpha\lambda_i[P_i]^{m_i}\prod_{j\neq i}(\lambda_j[P_j]^{m_j} + 1)}{\beta \prod_{j\neq i}(\mu_j[P_j]^{m_j} + 1) + \alpha\lambda_i[P_i]^{m_i}\prod_{j\neq i}(\lambda_j[P_j]^{m_j} + 1)} \tag{13}$$

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

3.5 Parameterization for inference

Parameters of equation (13) 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 (13), we obtain

$$k_{\text{on}} = \frac{k_0 + k_1\Phi([P_i]/s_i)^{m_i}}{1 + \Phi([P_i]/s_i)^{m_i}}$$

where

$$\Phi = \exp(\theta_i)\prod_{j\neq i}\frac{1 + \exp(\theta_j)([P_j]/s_j)^{m_j}}{1 + ([P_j]/s_j)^{m_j}}.$$

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 $y_j = [P_j]$ 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:

$$k_{\text{on},i}(y) = \frac{k_{0,i} + k_{1,i}\Phi_i(y)(y_i/s_{i,i})^{m_{i,i}}}{1 + \Phi_i(y)(y_i/s_{i,i})^{m_{i,i}}} \tag{14}$$

with

$$\Phi_i(y) = \exp(\theta_{i,i})\prod_{j\neq i}\frac{1 + \exp(\theta_{i,j})(y_j/s_{i,j})^{m_{i,j}}}{1 + (y_j/s_{i,j})^{m_{i,j}}}. \tag{15}$$

In our statistical framework, 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 $\theta = (\theta_{i,j}) \in \mathcal{M}_n(\mathbb{R})$, which is similar to the interaction matrix in usual gene network inference methods.

3.6 Explicit distribution for an auto-activation model

Here we derive the stationary distribution for a self-activating gene. For simplicity, we drop the i index. In this model, k_{off} is constant and we assume that there are some constants $\Phi \geqslant 0$, $m \geqslant 0$, $s > 0$ and $k_1 \gg k_0 > 0$ such that k_{on} has the form

$$k_{\text{on}}(y) = \frac{k_0 + k_1\Phi(y/s)^m}{1 + \Phi(y/s)^m}$$

so the stationary distribution can directly be used in the Hartree approximation of the network model (14), recalling that Φ has to be independent of the gene's own protein but can depend on others. Letting $c = (k_1 - k_0)/(md_1) > 0$, we are in the case of the explicit solution (8) with

$$\varphi(y) = c \log\left(y^{\frac{k_0}{d_1 c}} + \frac{\Phi}{s^m} y^{\frac{k_1}{d_1 c}} \right) + \frac{k_{\text{off}}}{d_1} \log(1-y)$$

so the protein distribution is

$$\underline{u}(y) = Z^{-1} y^{-1} \left(y^{\frac{k_0}{d_1 c}} + \frac{\Phi}{s^m} y^{\frac{k_1}{d_1 c}} \right)^c (1-y)^{\frac{k_{\text{off}}}{d_1} - 1}. \tag{16}$$

To get a fully explicit result, i.e. to compute Z, we shall assume that c is a positive integer. If it is not, one can get a satisfying approximation by taking $c = \lceil (k_1 - k_0)/(md_1) \rceil$. Then, expanding (16) using the binomial theorem, we obtain

$$Z = \sum_{r=0}^{c} \binom{c}{r} \mathrm{B}(a_r, b) (\Phi / s^m)^r$$

where $a_r = ((c-r)k_0 + rk_1)/(d_1 c)$ and $b = k_{\text{off}}/d_1$, and a probabilistic representation of $\underline{u}$ in terms of a mixture of Beta distributions:

$$\underline{u}(y) = \sum_{r=0}^{c} p_r f_r(y) \tag{17}$$

where $f_r(y) = y^{a_r - 1}(1-y)^{b-1} / \mathrm{B}(a_r, b)$ and $p_r = \binom{c}{r} \mathrm{B}(a_r, b)(\Phi/s^m)^r / Z$.

The dissociation constant s is clearly redundant with the input Φ. We fix the particular value

$$s = \left(\frac{\mathrm{B}(a_c, b)}{\mathrm{B}(a_0, b)} \right)^{\frac{1}{mc}} = \left(\frac{\mathrm{B}(k_1/d_1, k_{\text{off}}/d_1)}{\mathrm{B}(k_0/d_1, k_{\text{off}}/d_1)} \right)^{\frac{d_1}{k_1 - k_0}} \tag{18}$$

for which the arbitrary neutral case $\Phi = 1$ is "symmetric", i.e. $p_0 = p_c$. Note that s actually only depends on the fundamental parameters k_0, k_1, k_{off} and d_1 (and not on c nor m). Figure S2 shows some examples of the resulting distribution, which can be bimodal or not, depending on the value of c (or equivalently, m) when all other parameters are fixed.

4 EM algorithm for network inference

4.1 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:

$$p(\theta \,|\, x) = \int p(\theta, y \,|\, x)\, \mathrm{d}y \quad \text{where} \quad p(\theta, y \,|\, x) = p(y \,|\, \theta) p(x \,|\, y, \theta) \frac{p(\theta)}{p(x)}.$$

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

$$\theta \mapsto Q(\theta, \theta^t) = \int \ell_\theta(x, y) p(y \,|\, x, \theta^t)\, \mathrm{d}y. \tag{19}$$

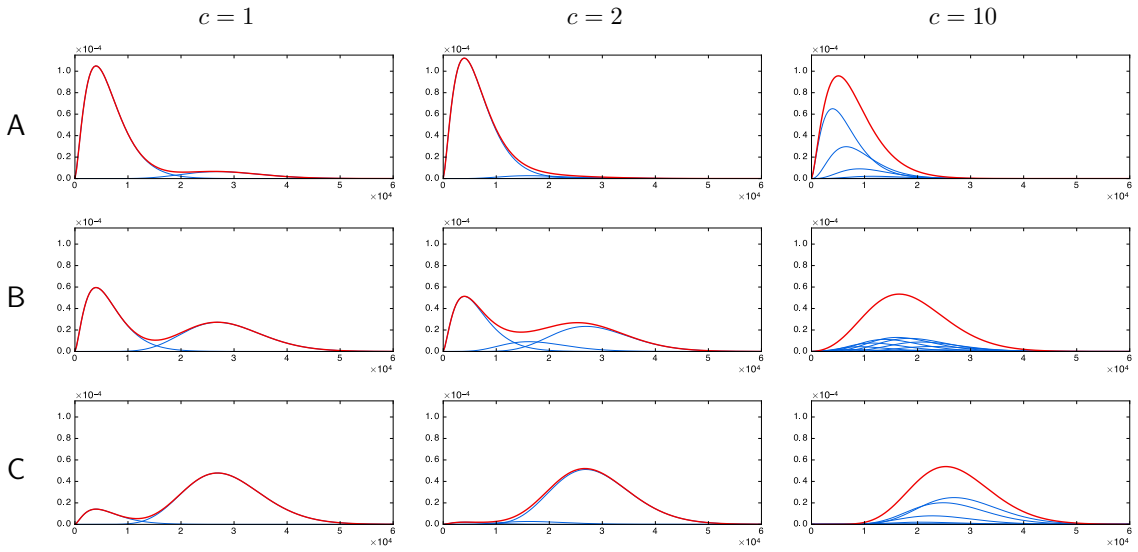


Figure S2: Protein stationary distributions (red curves) from the auto-activation model for different values of the input Φ, with s set as in (18). The blues curves indicate the underlying weighted Beta distributions in each mixture. (A) $\Phi = \exp(-2)$, (B) $\Phi = \exp(0) = 1$, (C) $\Phi = \exp(2)$. The distribution tends to be strongly bimodal for small c values, while large values make the distribution close to the unimodal no-feedback case (constant k_{on}). Parameters are $k_0 = 0.25$, $k_1 = 1.25$, $k_{\text{off}} = 7.5$, $d_1 = 0.1$ and, only used for scaling, $s_1 = 10$, $d_0 = 0.5$, $s_0 = 10^3$.

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 [2] 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 $\overline{Q}(\theta, \theta^t) = \int [\log p(y \,|\, \theta) + \log p(x \,|\, y, \theta)] p(y \,|\, x, \theta^t)\, \mathrm{d}y$ 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.

4.2 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:

$$p(\theta) \propto \exp\left(-\lambda \sum_{i \neq j} |\theta_{i,j}| - \lambda\alpha \sum_{i<j} |\theta_{i,j}\theta_{j,i}|\right) \tag{20}$$

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.

4.3 The algorithm in practice

As visible in (19), 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 [3]. 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 sections 4.1-4.2 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

$$\mathcal{F}(\mathbf{y}, \theta) = \ell(\mathbf{x}, \mathbf{y}, \theta) - g(\theta) \tag{21}$$

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:

$$\mathbf{y}^{t+1} = \arg\max_{\mathbf{y}}\{\mathcal{F}(\mathbf{y}, \theta^t)\} \tag{22}$$

$$\theta^{t+1} = \arg\max_{\theta}\{\mathcal{F}(\mathbf{y}^{t+1}, \theta)\} \tag{23}$$

The "approximate E step" (22) can be performed using a standard gradient method since u and v are smooth functions of y. The "penalized M step" (23) 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 $\nabla \log v$.

The formulas for u and v derived from the normalized model are given in the main text: they can be applied once the data has been normalized, i.e. after dividing each mRNA i level by $s_{0,i}/d_{0,i}$. In the bursty regime, this scale parameter is neither identifiable nor necessary. Indeed, as explained in the main text, the "Beta-like" distributions collapse to "Gamma-like" ones which we provide below, and for which the scale parameter is identifiable.

4.3.1 Likelihood form in the basic case

In the basic case (no self-interaction), we have:

$$\log(u(y,\theta)) = \sum_{i=1}^{n} (a_i(y,\theta)-1)\log(y_i) - b_i(y,\theta)y_i + a_i(y,\theta)\log(b_i(y,\theta)) - \log\Gamma(a_i(y,\theta))$$

and

$$\log(v(x,y,\theta)) = \sum_{i=1}^{n} (\widetilde{a}_i(y,\theta)-1)\log(x_i) - \widetilde{b}_i(y,\theta)x_i + \widetilde{a}_i(y,\theta)\log(\widetilde{b}_i(y,\theta)) - \log\Gamma(\widetilde{a}_i(y,\theta))$$

where $a_i = k_{\text{on},i}/d_{1,i}$, $b_i = (d_{0,i}/s_{1,i}) \times (k_{\text{off},i}/s_{0,i})$, $\widetilde{a}_i = k_{\text{on},i}/d_{0,i}$ and $\widetilde{b}_i = k_{\text{off},i}/s_{0,i}$.

4.3.2 Likelihood form in the auto-activation case

In the auto-activation case (section 3.6), the formula for $\log v$ is the same but $\log u$ is given by

$$\log(u(y,\theta)) = \sum_{i=1}^{n} \log\left(\sum_{r=0}^{c_i} w_{i,r}(y,\theta) {y_i}^{a_{i,r}-1} e^{-b_i y_i}\right) - \log\left(\sum_{r=0}^{c_i} w_{i,r}(y,\theta)\Gamma(a_{i,r}){b_i}^{-a_{i,r}}\right)$$

where $c_i = \lceil (k_{1,i}-k_{0,i})/(d_{1,i}m_{i,i}) \rceil$, $a_{i,r} = ((c_i-r)k_{0,i} + rk_{1,i})/(d_{1,i}c_i)$ and $w_{i,r} = \binom{c_i}{r}(\Phi_i/s_{i,i}^{m_{i,i}})^r$.

4.3.3 Gradients

Explicit computation of the gradients is then straightforward (e.g. with $k_{\text{off},i}$ constant and $k_{\text{on},i}$, Φ_i defined by (14)-(15)) but leads to cumbersome formulas: we implemented them in Scilab and the code is available upon request.

4.4 Proximal gradient method

Here we recall a standard proximal gradient method [4] to solve the M step (23) 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

$$\theta^{(k+1)} = \text{prox}_\gamma\left(\theta^{(k)} + \gamma\nabla_\theta \ell(\mathbf{x},\mathbf{y},\theta^{(k)})\right)$$

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

$$\text{prox}_\gamma(\tau) = \arg\min_{\theta\in\Theta}\left\{ g(\theta) + \frac{1}{2\gamma}\sum_{i\neq j}(\theta_{i,j}-\tau_{i,j})^2 \right\}.$$

In fact, for any $i,j \in \{1,\dots,n\}$ 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

$$\text{prox}_\gamma(\tau_1,\tau_2) = \arg\min_{(\theta_1,\theta_2)\in\mathbb{R}^2}\left\{ \lambda\left(|\theta_1| + |\theta_2| + \alpha|\theta_1\theta_2|\right) + \frac{1}{2\gamma}\left((\theta_1-\tau_1)^2 + (\theta_2-\tau_2)^2\right)\right\}$$

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 $\text{prox}_\gamma(\tau)$. Then, letting $\varepsilon = \lambda\gamma$ and assuming γ small enough such that $\alpha\varepsilon < 1$, we obtain

$$\text{prox}_\gamma(\tau_1,\tau_2) = \frac{1}{1-(\alpha\varepsilon)^2}(h_1,h_2)$$

with 9 cases for the value of (h_1,h_2) depending on (τ_1,τ_2), given by:

$$
1.\ \begin{cases} \tau_1 > \varepsilon \\ \tau_1 > \varepsilon(1+\alpha(\tau_2-\varepsilon)) \\ \tau_2 > \varepsilon(1+\alpha(\tau_1-\varepsilon)) \end{cases} \Rightarrow \begin{cases} h_1 = \tau_1 - \varepsilon(1+\alpha(\tau_2-\varepsilon)) \\ h_2 = \tau_2 - \varepsilon(1+\alpha(\tau_1-\varepsilon)) \end{cases}
$$

$$
2.\ \begin{cases} \tau_1 > \varepsilon \\ |\tau_2| \leqslant \varepsilon(1+\alpha(\tau_1-\varepsilon)) \end{cases} \Rightarrow \begin{cases} h_1 = \tau_1 - \varepsilon \\ h_2 = 0 \end{cases}
$$

$$
3.\ \begin{cases} \tau_1 > \varepsilon \\ \tau_1 > \varepsilon(1+\alpha(-\tau_2-\varepsilon)) \\ \tau_2 < -\varepsilon(1+\alpha(\tau_1-\varepsilon)) \end{cases} \Rightarrow \begin{cases} h_1 = \tau_1 - \varepsilon(1+\alpha(-\tau_2-\varepsilon)) \\ h_2 = \tau_2 + \varepsilon(1+\alpha(\tau_1-\varepsilon)) \end{cases}
$$

$$
4.\ \begin{cases} |\tau_1| \leqslant \varepsilon(1+\alpha(-\tau_2-\varepsilon)) \\ \tau_2 < -\varepsilon \end{cases} \Rightarrow \begin{cases} h_1 = 0 \\ h_2 = \tau_2 + \varepsilon \end{cases}
$$

$$
5.\ \begin{cases} \tau_1 < -\varepsilon \\ \tau_1 < -\varepsilon(1+\alpha(-\tau_2-\varepsilon)) \\ \tau_2 < -\varepsilon(1+\alpha(-\tau_1-\varepsilon)) \end{cases} \Rightarrow \begin{cases} h_1 = \tau_1 + \varepsilon(1+\alpha(-\tau_2-\varepsilon)) \\ h_2 = \tau_2 + \varepsilon(1+\alpha(-\tau_1-\varepsilon)) \end{cases}
$$

$$
6.\ \begin{cases} \tau_1 < -\varepsilon \\ |\tau_2| \leqslant \varepsilon(1+\alpha(-\tau_1-\varepsilon)) \end{cases} \Rightarrow \begin{cases} h_1 = \tau_1 + \varepsilon \\ h_2 = 0 \end{cases}
$$

$$
7.\ \begin{cases} \tau_1 < -\varepsilon \\ \tau_1 < -\varepsilon(1+\alpha(\tau_2-\varepsilon)) \\ \tau_2 > \varepsilon(1+\alpha(-\tau_1-\varepsilon)) \end{cases} \Rightarrow \begin{cases} h_1 = \tau_1 + \varepsilon(1+\alpha(\tau_2-\varepsilon)) \\ h_2 = \tau_2 - \varepsilon(1+\alpha(-\tau_1-\varepsilon)) \end{cases}
$$

$$
8.\ \begin{cases} |\tau_1| \leqslant \varepsilon(1+\alpha(\tau_2-\varepsilon)) \\ \tau_2 > \varepsilon \end{cases} \Rightarrow \begin{cases} h_1 = 0 \\ h_2 = \tau_2 - \varepsilon \end{cases}
$$

$$
9.\ \begin{cases} |\tau_1| \leqslant \varepsilon \\ |\tau_2| \leqslant \varepsilon \end{cases} \Rightarrow \begin{cases} h_1 = 0 \\ h_2 = 0 \end{cases}
$$

These 9 cases form a partition of $\mathbb{R}^2$ and are represented in Figure S3. One can check that the case $\alpha = 0$ collapses to the usual proximal operator associated with lasso penalization.

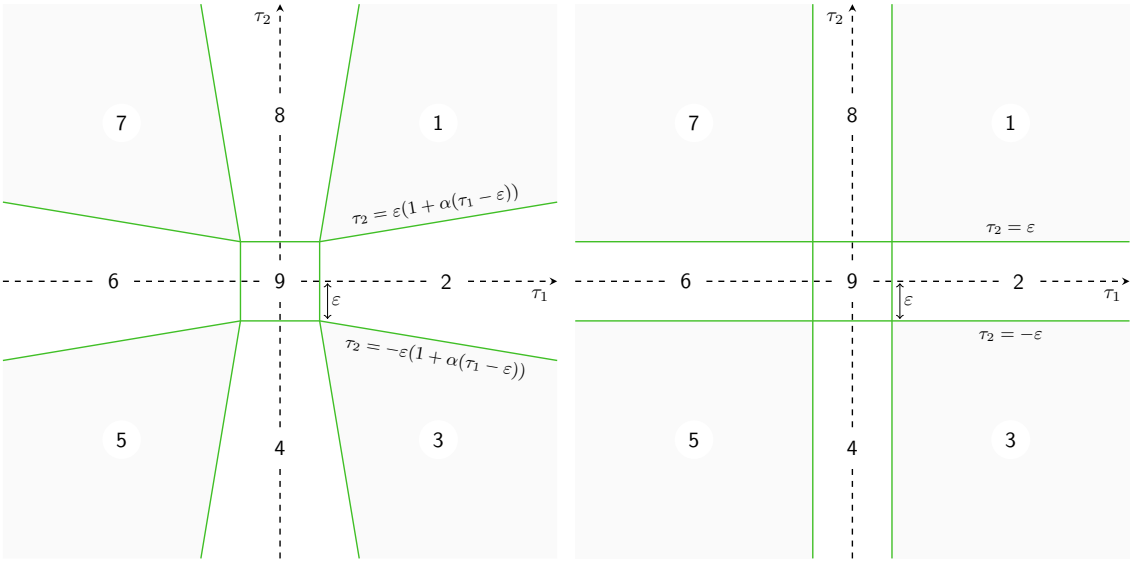


Figure S3: Partition of $\mathbb{R}^2$ associated with the proximal operator, for $\alpha > 0$ (left) and $\alpha = 0$ (right). Gray areas correspond to a usual gradient and white areas correspond to a threshold.

To obtain the results of Fig 8, we used $\lambda = 10$, $\alpha = 5$ and $\gamma = 10^{-4}$. In a broader context, one may use standard cross-validation to derive appropriate values for λ and α.

5 Dealing with real data

In this section, we propose a pre-processing phase which would be required in order to apply our network inference method to real data. The first step ensures that the approximate likelihood is well-defined, while the second step consists in estimating the basal parameters appearing in functions $k_{\mathrm{on},i}$, $k_{\mathrm{off},i}$. Please note that inferring real networks from real data is beyond the scope of this paper and will be the subject of future papers.

5.1 Spreading zeros

The likelihood does not accept exact zeros (cf. section 4.3). This is not a problem with continuous-type data (for instance, based on fluorescence measurements), but it becomes one when dealing with counts (e.g. RNA-Seq). We propose to replace such zeros with relevant positive values. Recall that the PDMP focuses on the promoter and neglects the local molecular noise at the mRNA level. It is therefore natural to consider that, given a value $M > 0$ of mRNA level in the PDMP, the actual number m of molecules in the cell is drawn from the Poisson distribution $\mathcal{P}(M)$. Then, a possible way to replace zeros is to go backwards, i.e. to draw a value M from the PDMP distribution conditioned to $m = 0$. Namely, we propose the following procedure to be applied independently for each gene:

1. Infer a gamma distribution $\gamma(a, b)$ (as a local approximation of the PDMP) from the whole data (possibly at a given time-point) using the standard method of moments;
2. Replace zeros with independent samples from the distribution $\gamma(a, b + 1)$, conditioned to be smaller than the smallest positive value that was measured.

This procedure ensures that zeros are replaced with very small values and that no artificial correlation is introduced. The distribution $\gamma(a, b+1)$ comes from the fact that, if $\mathcal{L}(M) = \gamma(a, b)$ and $\mathcal{L}(m|M) = \mathcal{P}(M)$, then a simple computation gives $\mathcal{L}(M|m = 0) = \gamma(a, b + 1)$.

5.2 Estimating basal parameters

Here we describe a heuristic method to estimate the model-specific parameters (i.e. everything but the matrix θ) when they cannot be measured through *ad hoc* experiments, in the case of the auto-activation form (14)-(15). Once again we refer to section 4.3. Note that for the mechanistic approach to be relevant, one should know at least the ratio $d_{0,i}/d_{1,i}$, which can be obtained by measuring mRNA and protein half-lives. When even this is unavailable, we propose to use the default value $d_{0,i}/d_{1,i} = 5$ (mean value derived from the literature, cf. main text).

The main idea consists in noticing that, when protein i is described by the auto-activation model (14)-(15) (thus following the distribution (17)), mRNA i in quasi-steady state happens to be well described by the same distribution class as (17), with the same $m_{i,i}$ and other parameters being divided by $d_{0,i}/d_{1,i}$. More precisely, we perform the following steps:

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 for instance using an EM algorithm for each value of $\widetilde{c}_i$ in some range (e.g. $\widetilde{c}_i = 1, 2, \ldots, 10$), 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 of (18) 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 regarding section 4.3.2, 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 $1/b_i$.

3. Last step consists in extrapolating $m_{i,i}$ and $s_{i,i}$ to the remaining unknown parameters $m_{i,j}$ and $s_{i,j}$ (describing how gene j influences gene i). Since the crucial point is their coherence with respect to the range of protein j, a relevant choice without additional knowledge is, for all $i \neq j$,

$$m_{i,j} = m_{j,j} \quad \text{and} \quad s_{i,j} = s_{j,j}.$$

6 Parameter values

6.1 Models

Table S1: General parameters used in the examples. The $s_{i,j}$ correspond to the normalized model: counterparts in absolute protein numbers are $\overline{s}_{i,j} = s_{i,j} \times (s_0 s_1)/(d_0 d_1) = 2 \times 10^3$ for $i \neq j$ and $\overline{s}_{i,i} = s_{i,i} \times (s_0 s_1)/(d_0 d_1) = 1.9 \times 10^4$.

Parameter	Value	Units
s_0	10^3	$\text{mRNA} \cdot \text{h}^{-1}$
s_1	10	$\text{protein} \cdot \text{h}^{-1} \cdot \text{mRNA}^{-1}$
d_0	0.5	h^{-1}
d_1	0.1	h^{-1}
k_0	0.34	h^{-1}
k_1	2.15	h^{-1}
k_{off}	10	h^{-1}
$m_{i,j}$	2 for $i \neq j$	–
$m_{i,i}$	2 (Fig 5, 6) or 3 (Fig 8)	–
$s_{i,j}$	0.01 for $i \neq j$	proteins (normalized)
$s_{i,i}$	0.095 from eq. (18)	proteins (normalized)

Table S2: Network parameters used in the examples.

Fig 5, 6	$\theta_{1,1}$	$\theta_{1,2}$	$\theta_{2,1}$	$\theta_{2,2}$
	4	−8	−8	4
Fig 8	$\theta_{1,1}$	$\theta_{1,2}$	$\theta_{2,1}$	$\theta_{2,2}$
1	0	0	0	0
2	0	0	1	0
3	0	1	0	0
4	−0.1	1	1	−0.1
5	0	0	−1	0
6	0	−1	0	0
7	0	−1	−1	0

6.2 Results

Table S3: Inferred network parameters used to generate Fig 8b. Each row refers to one of the ten datasets generated for testing. Colors indicate whether the parameters represent the correct topology (blue) or another one (orange) regarding the true networks (cf. Table S2).

Network 1		
	$\theta_{1,2}$	$\theta_{2,1}$
1	0	0
2	0	0
3	0	0
4	0	0
5	0.10	0
6	0	0
7	0	0
8	0	0
9	0	0
10	0	0

Network 2		
	$\theta_{1,2}$	$\theta_{2,1}$
1	0	0.13
2	0	0.18
3	0	0.14
4	0	0.17
5	0	0
6	0	0.20
7	0	0.18
8	0	0.11
9	0	0.15
10	0	0

Network 3		
	$\theta_{1,2}$	$\theta_{2,1}$
1	0	0
2	0.12	0
3	0.26	0.10
4	0.19	0
5	0.21	0
6	0.36	0.15
7	0.11	0
8	0.11	0
9	0.19	0
10	0.12	0

Network 4		
	$\theta_{1,2}$	$\theta_{2,1}$
1	0.28	0.22
2	0.25	0.15
3	0.23	0.21
4	0.20	0.15
5	0.18	0.23
6	0.23	0.27
7	0.17	0.16
8	0.14	0.13
9	0.29	0.21
10	0.24	0.21

Network 5		
	$\theta_{1,2}$	$\theta_{2,1}$
1	0	0
2	0	−0.12
3	0	−0.10
4	0	−0.16
5	−0.10	−0.21
6	0	−0.18
7	0	−0.13
8	0	−0.09
9	0	0
10	0	−0.18

Network 6		
	$\theta_{1,2}$	$\theta_{2,1}$
1	−0.14	0
2	−0.11	0
3	−0.13	0
4	−0.15	0
5	−0.21	0
6	−0.16	0
7	−0.29	−0.09
8	0	0
9	−0.13	0
10	−0.21	0

Network 7		
	$\theta_{1,2}$	$\theta_{2,1}$
1	−0.20	−0.24
2	−0.34	−0.34
3	−0.17	−0.14
4	−0.23	−0.20
5	−0.17	−0.18
6	−0.19	−0.16
7	−0.21	−0.21
8	−0.30	−0.31
9	−0.11	−0.13
10	−0.11	−0.17

Table S4: Example of inferred networks in the presence of dropouts (30% of the whole dataset) generated by applying a Poisson noise and then a threshold to the "perfect" data. Such zeros where replaced using the procedure described in section 5.1 before inferring the networks.

Network 1		
	$\theta_{1,2}$	$\theta_{2,1}$
1	0	0
2	0	0.10
3	0	0
4	0	0
5	0.10	0
6	0	0
7	0.09	0.16
8	0	0
9	0	0
10	0	0

Network 2		
	$\theta_{1,2}$	$\theta_{2,1}$
1	0	0.27
2	0	0.16
3	0	0.31
4	0	0.44
5	0	0.27
6	0	0.55
7	0	0.23
8	0	0.21
9	0	0.36
10	0	0

Network 3		
	$\theta_{1,2}$	$\theta_{2,1}$
1	0.31	0
2	0.23	0
3	0.32	0.12
4	0.30	0
5	0.39	0.11
6	0.40	0.16
7	0.19	0.11
8	0.14	0
9	0.35	0.16
10	0.19	0

Network 4		
	$\theta_{1,2}$	$\theta_{2,1}$
1	0.43	0.41
2	0.44	0.26
3	0.36	0.28
4	0.44	0.23
5	0.38	0.51
6	0.42	0.48
7	0.48	0.34
8	0.36	0.32
9	0.59	0.43
10	0.41	0.34

Network 5		
	$\theta_{1,2}$	$\theta_{2,1}$
1	0	0
2	0	−0.26
3	−0.09	−0.21
4	−0.11	−0.35
5	−0.30	−0.47
6	0	−0.32
7	0	−0.22
8	0	−0.19
9	0	−0.27
10	0	−0.38

Network 6		
	$\theta_{1,2}$	$\theta_{2,1}$
1	−0.18	0
2	−0.25	0
3	−0.28	0
4	−0.25	0
5	−0.34	0
6	−0.26	0
7	−0.47	−0.18
8	−0.09	0
9	−0.22	0
10	−0.38	−0.12

Network 7		
	$\theta_{1,2}$	$\theta_{2,1}$
1	−0.22	−0.24
2	−0.5	−0.50
3	−0.25	−0.21
4	−0.26	−0.22
5	−0.12	−0.14
6	−0.20	−0.18
7	−0.30	−0.29
8	−0.44	−0.45
9	−0.15	−0.18
10	−0.11	−0.17

References

[1] A. M. Walczak, M. Sasai, and P. G. Wolynes, "Self-consistent proteomic field theory of stochastic gene switches," *Biophysical Journal*, vol. 88, pp. 828–850, 2005.

[2] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul, "An Introduction to Variational Methods for Graphical Models," *Machine Learning*, vol. 37, no. 2, pp. 183–233, 1999.

[3] G. Celeux and G. Govaert, "A classification EM algorithm for clustering and two stochastic versions," Research Report 1364, INRIA, 1991.

[4] N. Parikh and S. Boyd, "Proximal Algorithms," *Foundations and Trends in Optimization*, vol. 1, no. 3, pp. 123–231, 2013.

2.3 Article 3 : WASABI : a dynamic iterative framework for gene regulatory network inference

L'approche précédente a permis de définir un modèle mécaniste de RRG dit de PDMP couplés. Cependant, l'inférence de RRG à partir de ce modèle par une approche purement mathématique se heurte encore aux limites typiques de l'exercice que j'ai présentées dans l'introduction. En reprenant le modèle mécaniste, j'ai décidé d'aborder la problématique de l'inférence sous un angle différent, en utilisant des concepts dynamiques de traitement du signal issus de l'ingénierie, avec l'objectif de surmonter plusieurs de ces limites.

Je vais commencer par présenter quelques principes du traitement du signal qui nous serons utiles pour la compréhension de la notion de "vagues" d'expression présentée dans l'article 3.

2.3.1 Le processus de différenciation vu comme un processus dynamique de traitement du signal par les RRG

Cette partie est une brève introduction aux principes du traitement du signal appliqués aux RRG. Pour les lecteurs qui souhaitent avoir plus de détails sur ce sujet nous vous conseillons la lecture de ce livre [109] qui est une excellente introduction.

Dans le domaine du traitement du signal on définit un système par ses capacités à traiter des signaux (comme le filtrage), à les analyser (opérations logiques) et à les interpréter (comparaison à un contexte). Un tel système peut être modélisé par une EDO comme nous l'avons déjà présenté dans la partie **2.3.5.1**. En reprenant les notations de l'équation 5, nous allons maintenant montrer que les réactions biochimiques de création et de dégradation des molécules se comportent comme un filtre passe bas du 1er ordre et induisent un délai dans la transmission de l'information comme le résume la figure 4.

Les RRG font intervenir la production et la dégradation d'ARNm (noté M) et de protéine (noté P). Dans le modèle classique déterministe ou un stimulus (noté U) induit l'expression d'un gène on peut écrire la dynamique de ce système par les équations différentielles suivantes (dans ce cas le vecteur d'état est défini par $X = [M, P]$) :

$$\begin{cases} \dfrac{dM}{dt} = s_0 U - d_0 M \\ \dfrac{dP}{dt} = s_1 M - d_1 P \end{cases} \tag{7}$$

Pour simplifier ce qui suit on ne considère que l'équation relative aux protéines. En passant dans le domaine fréquentiel et en utilisant la transformée de Laplace, où une dérivation revient à multiplier par $j\omega$ (avec ω la pulsation), cette équation devient :

$$j\omega P = s_1 M - d_1 P \tag{8}$$

On en déduit alors la fonction de transfert entre les ARN et les protéines :

$$\frac{P}{M} = \frac{s_1}{d_1} \frac{1}{1 + j\frac{\omega}{d_1}} \tag{9}$$

On retrouve la forme typique de la fonction de transfert $H(\omega)$ d'un filtre passe bas du 1er ordre de pulsation de coupure ω_c qui est rappelé ci-dessous (on donne également la phase associé $\Phi(\omega)$ qui correspond au retard de phase engendré par le filtre) :

$$\begin{aligned} H(\omega) &= \frac{K}{1 + j\frac{\omega}{\omega_c}} \\ \Phi(\omega) &= -\arctan(\frac{\omega}{\omega_c}) \end{aligned} \tag{10}$$

On en déduit que le processus de production/dégradation des protéines à partir des ARNm est équivalent à un filtre passe bas du 1er ordre de gain $K = \frac{s_1}{d_1}$ et de pulsation de coupure $\omega_c = d_1$. Les filtres passe bas ont la propriété d'atténuer fortement les composantes du signal de fréquence supérieure à la pulsation de coupure. Une autre propriété, qui nous intéresse plus, est le délai $\tau(\omega)$ induit par le filtre sur un signal de pulsation ω qui est donné par :

$$\tau(\omega) = -\frac{\Phi(\omega)}{\omega} = \frac{\arctan(\frac{\omega}{\omega_c})}{\omega} \tag{11}$$

Dans le cas ou $\omega << \omega_c$ on peut approximer le délai par :

$$\tau(\omega) = \frac{1}{\omega_c} = \frac{1}{d_1} \tag{12}$$

Ainsi, l'information contenue par les ARNm sera propagée au niveau des protéines avec un délai correspondant à l'inverse du taux de dégradation des protéines. Le même raisonnement

peut être mené entre le stimulus et les ARNm. De fait, le délai total entre le stimulus et les protéines peut s'approximer par la somme des inverses des taux de dégradation des ARNm et des protéines.

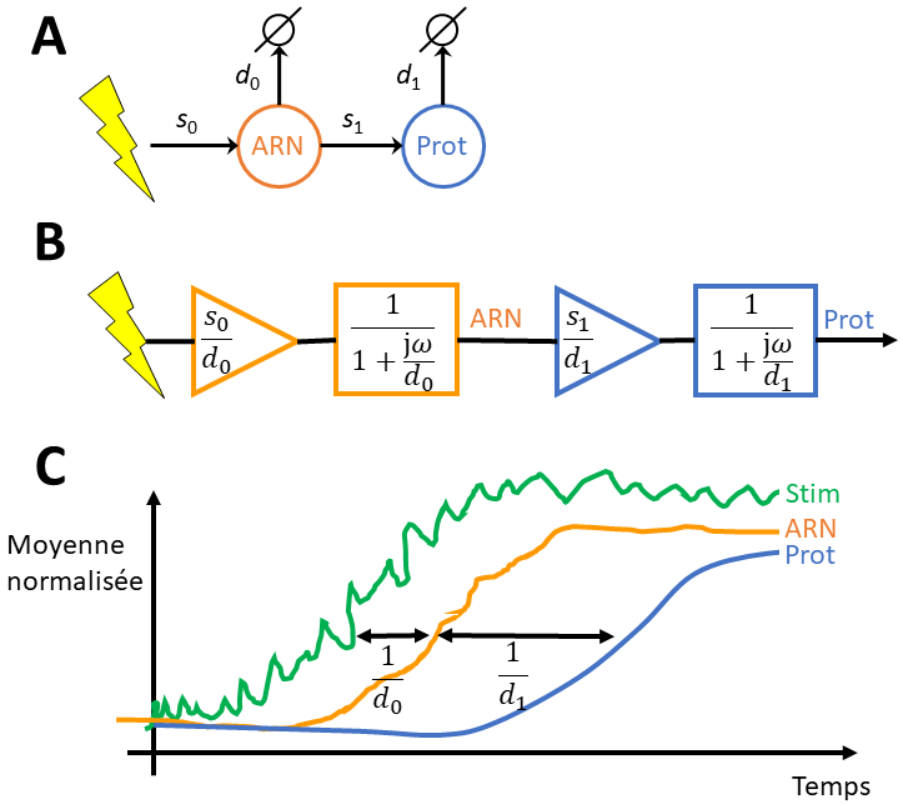


Figure 4 – L'expression génétique vu comme un processus de traitement du signal
A) Représentation classique de l'expression d'un gène à l'échelle de la population induit par un stimulus. Les paramètres s0 et s1 représentent respectivement les taux de transcription et traduction. Les paramètres d0 et d1 représentent respectivement les taux de dégradation des ARN et des protéines. B) Représentation du même processus en utilisant les notions de traitement du signal. Le signal passe par les étapes successives d'amplification et de filtrage. Les triangles représentent une amplification avec le gain correspondant. Les carrés représentent des filtres passe-bas du 1er ordre avec la fonction de transfert associée. C) Exemple de trace temporelle typique de la moyenne des ARN et protéines suite à une stimulation bruitée. La moyenne des ARN est moins bruitée que le stimulus et retardée de $\frac{1}{d_0}$. Les protéines filtrent le signal de l'ARN avec un délai de $\frac{1}{d_1}$. Dans notre modèle biologique, les protéines ont généralement des demie-vie d'environ 20h contrairement aux ARN qui ont environ 4h de demie-vie.

2.3.2 Principaux résultats de l'article 3

A partir du concept de vagues d'expression validé expérimentalement dans l'article 1, j'ai développé un algorithme baptisé WASABI pour "Waves Analysis Based Inference" (figure 1 de l'article 3). Le principe repose sur le fait que les gènes sont activés les uns après les autres suivant la règle que la cause précède l'effet. On peut alors inférer les causalités du RRG par étape itérative en ajoutant les gènes un par un, suivant l'ordre chronologique de régulation. Cette méthode itérative permet de découper et paralléliser la problématique d'inférence de RRG. Une étape préliminaire d'estimation des paramètres individuels de chaque gène (comme les demi-vies des ARN et protéines) ainsi que leur temps de régulation est nécessaire.

Cet algorithme est dans un premier temps appliqué et validé pour l'inférence de RRG *in silico* composés de 6 gènes avec différentes topologies. Les données expérimentales sont simulées à l'aide du modèle de PDMP couplés développé dans l'article 2. Après 24h environ de calcul sur 400 machines en parallèle, WASABI retourne plusieurs dizaines de candidats dont le vrai RRG (figure 3 et 4 de l'article 3). Les boucles d'autoactivation et les rétroaction négatives présentes dans les RRG ont bien été inférées, ce qui montre la capacité de WASABI à inférer des causalités circulaires.

L'algorithme est ensuite appliqué sur les données *in vitro* d'expression en cellule unique issues de l'article 1, mais également à partir de données de cinétiques d'inhibition de la transcription en population pour estimer les demi-vies des ARN ainsi qu'une cinétique en protéomique sur population [110] pour estimer les paramètres relatifs aux protéines. Après 16 jours de calculs sur 400 machines 364 RRG candidats sont générés avec des topologies similaires (figure 6 de l'article 3) caractérisées par :

— un rôle central du stimulus

— une majorité d'inhibition par le stimulus sur ses cibles

— l'absence de "hub"

— la présence de plusieurs "incoherent feedforward loop"

— une profondeur de réseau limitée

— une forte proportion de boucles positives

Nous reviendrons en détails sur l'interprétation de ces résultats dans de la discussion de cette thèse.

2.3.3 Principales conclusions de l'article 3

Ces résultats montrent que l'algorithme WASABI proposé ici permet de surmonter quelques limitations comme :

— l'inférence de causalité même circulaire

— la génération de RRG candidats avec des topologies explicites qui incluent le stimulus et qui peuvent être simulées

— la capacité d'intégrer de la régulation post-traductionnelle (observée pour la moitié des gènes) grâce à l'analyse des données de protéomique

— la flexibilité et la conjuration de la "malédiction de la combinatoire" grâce à l'approche itérative

2.3.4 Article 3

Déposé en archive BioRxiv, en cours de revue dans le journal BMC Systems Biology.

URL : https ://www.biorxiv.org/content/early/2018/04/04/292128.abstract

e

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

Background: 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.

Results: 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 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.

Conclusions: Together, these results demonstrate WASABI versatility and ability to tackle some general gene regulatory networks inference issues. It is our hope that WASABI will prove useful in helping biologists to fully exploit the power of time-stamped single-cell data.

Keywords: Single-cell transcriptomics, Gene network inference, Multiscale modelling, proteomic, high parallel computing, T2EC, erythropoiesis.

Background

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 stands for "WAveS Analysis Based Inference". It 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

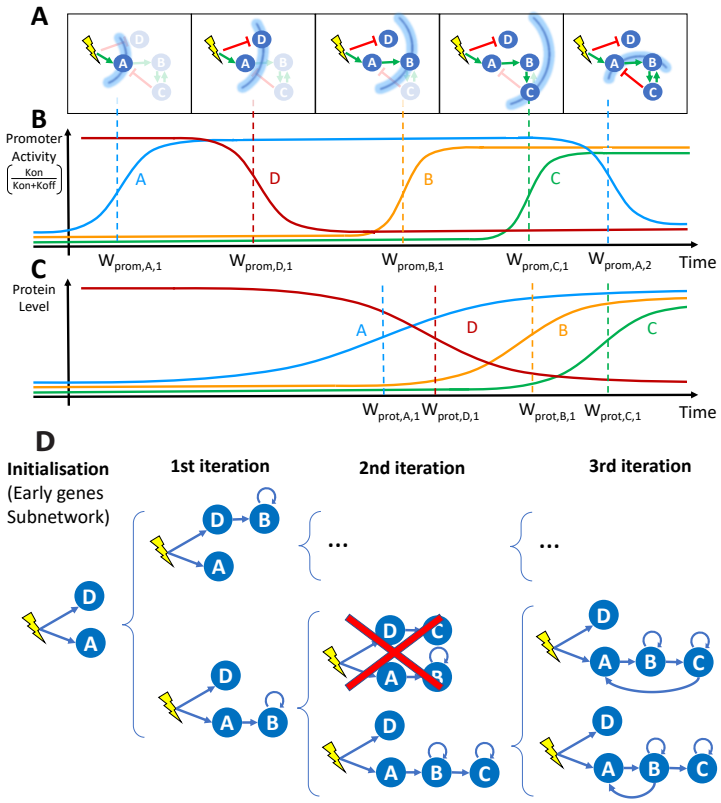


Fig 1. WASABI at a glance. A) Schematic view of a GRN: the stimulus is represented by a yellow flash, genes by blue circles and interactions by green (activation) or red (inhibition) arrows. The stimulus-induced information propagation is represented by blue arcs corresponding to wave times. Genes and 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 $k_{\text{on}}/(k_{\text{on}} + k_{\text{off}})$ 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.

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 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.

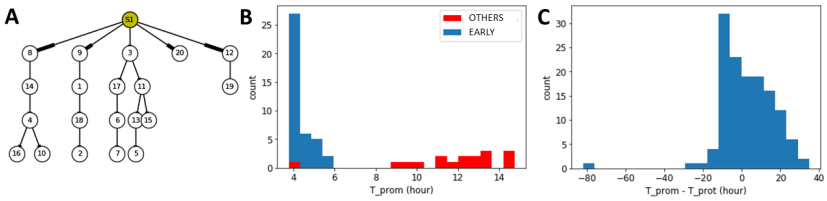


Fig 2. Cascade *in silico* GRN A) Cascade GRN types are generated to study wave dynamics. Genes correspond to *in vitro* ones with their estimated parameters. S1 corresponds to stimulus. Genes are identified by our list gene ID. B) Based on 10 *in silico* GRN we compare promoter wave time of early genes (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.

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 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.

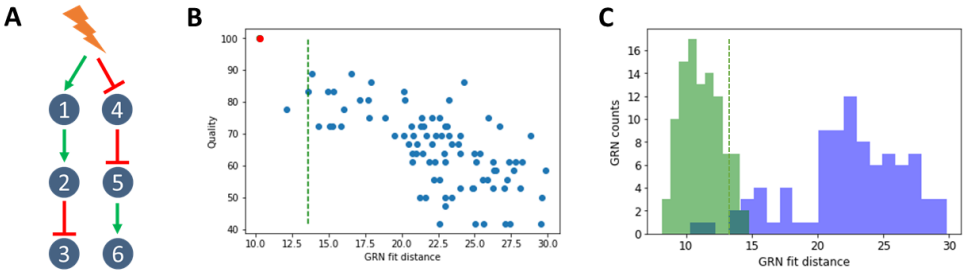


Fig 3. *In silico* cascade GRN inference A) The cascade GRN. Genes parameters were taken from *in vitro* estimations to mimic realistic behavior. Experimental data were generated to obtain time courses of transciptomic data, at single-cell and population scale, and also proteomic data at population scale. 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.

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

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

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).

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 [**?**] 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

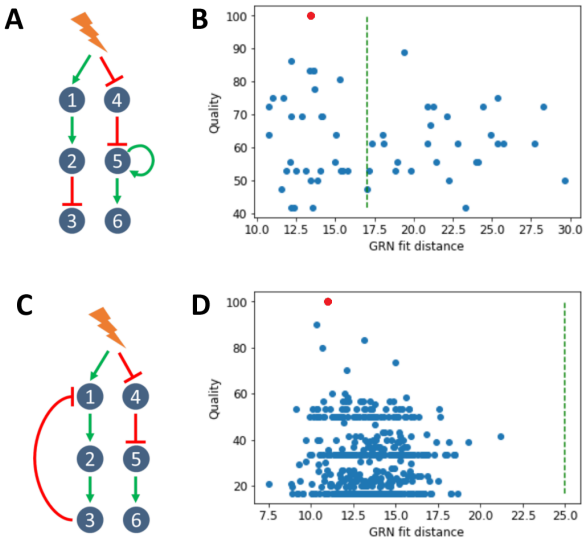


Fig 4. *In silico* GRN with feedbacks A) Addition of one positive feedback onto the cascade GRN. B) WASABI was run to infer *in silico* cascade GRN with a positive feedback and generated 59 candidates, 31 of which having an acceptable fit distance. See legend to Fig 3-B for details. C) Addition of one 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.

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.

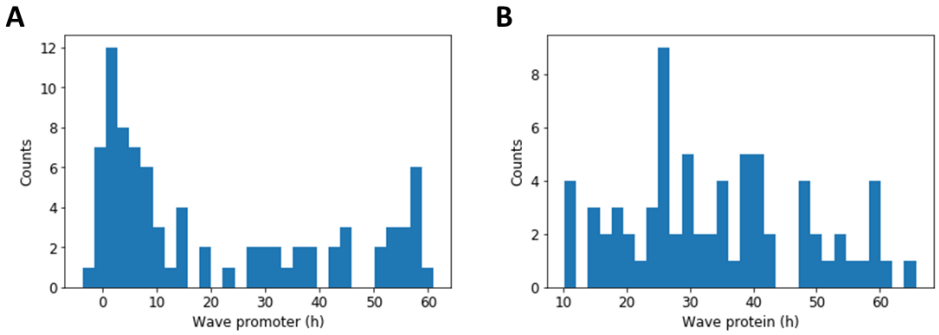


Fig 5. Promoter and protein wave time distributions. Distribution of *in vitro* promoter (A) and protein (B) wave times for all genes estimated from RNA and proteomic data at population scale. Counts represent number of genes. Note: a gene can have several waves for its promoter or protein.

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*

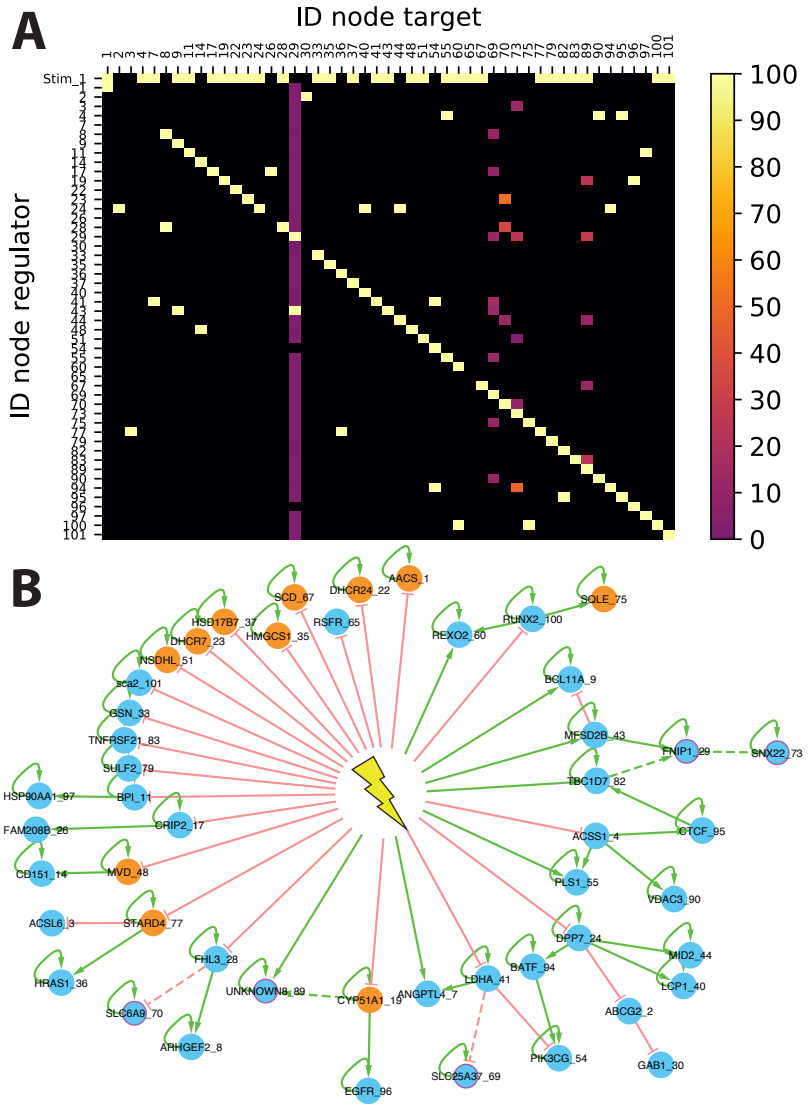


Fig 6. Inference from *in vitro* data A) *In vitro* interaction consensus matrix. Each square in the matrix represents either the absence of any interaction, in black, or the presence of an interaction, the frequency of which is color-coded, between the considered regulator ID (row) and regulated gene 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.

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.

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 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 totally inferred (Fig3) with a good 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.

Conclusions

Inferring the structure of GRN is an inverse problem which has occupied the systems biology community for decades. This last few years, with the arrival of single cell transcriptomic data, many GRN inference algorithms based on the analysis of these data have been developed. Despite their contributions and successes, these approaches are confronted to some limitations such as:

- restriction to correlations which impairs predictive ability
- restriction to transcription factors to target gene interactions
- mono-data type, namely transcriptomic, ignoring protein level regulation
- biological over-simplifying assumptions induced by mathematical tools

Our work aims to provide a significant innovation in GRN inference problem to tackle these issues. We propose 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. GRN candidates were generated with a very interesting common topology which stands apart from typical literature and which is biologically relevant regarding several aspects as the very central role of the stimulus, the absence of "hub genes", the limitation in network depth and the presence of many auto-activation loops.

Together, these results demonstrate WASABI ability to tackle some general GRN inference issues:

- inference of causalities even in case of feedbacks
- definition of functional interactions underlying indirect regulations as post-translational regulation or nuclear translocation
- capability to integrate proteomic data to reproduce translational or post-translational regulation (observed in 50% of our genes)
- versatility and computational tractability using HPC facilities enabled by WASABI original iterative process

We believe that WASABI should be of great interest for biologists interested in GRN inference, and beyond for those aiming at a dynamical network view of their biological processes. We are convinced that this could really advance the field, opening an entire new way of analyzing single cell data for GRN inference.

List of abbreviations

- WASABI = WAveS Analysis Based Inference
- GRN = Gene Regulatory Network
- TF = Transcription Factor
- DOE = Design Of Experiments
- HPC = High Parallel Computing

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.

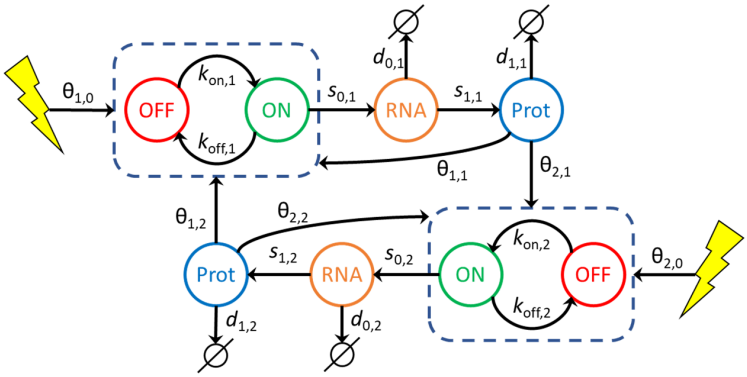


Fig 7. GRN mechanistic and stochastic model. Our GRN model is composed of coupled piecewise deterministic Markov processes. In this example 2 genes are coupled. A gene i is represented by its promoter state (dashed box) which can switch randomly from ON to OFF, and OFF to ON, respectively at $k_{\mathrm{on},i}$ and $k_{\mathrm{off},i}$ 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 $k_{\mathrm{on},i}$ and $k_{\mathrm{off},i}$ 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 $k_{\mathrm{on},i}$ and $k_{\mathrm{off},i}$ switching rates with interaction parameter $\theta_{i,0}$.

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 following equation, together with notations that will be extensively used throughout this article.

$$\begin{cases} E_i(t) : 0 \xrightarrow{k_{\mathrm{on}}} 1, \quad 1 \xrightarrow{k_{\mathrm{off}}} 0 \\ M_i'(t) = s_{0,i} E_i(t) - d_{0,i} M_i(t) \\ P_i'(t) = s_{1,i} M_i(t) - d_{1,i} P_i(t) \end{cases} \tag{1}$$

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 $-\theta_{i,j}$):

$$k_{\mathrm{on}}(P,Q) = \frac{k_{\mathrm{on\_min},i} + k_{\mathrm{on\_max},i}\beta_i \Phi_i(P,Q)}{1 + \beta_i \Phi_i(P,Q)} \tag{2}$$

$$\Phi_i(P,Q) = \frac{1 + e^{\theta_{i,0}} Q}{1+Q} \prod_{j=1}^{G} \frac{1 + e^{\theta_{i,j}} \left(\frac{P_j}{H_j}\right)^{\gamma}}{1 + \left(\frac{P_j}{H_j}\right)^{\gamma}} \tag{3}$$

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.

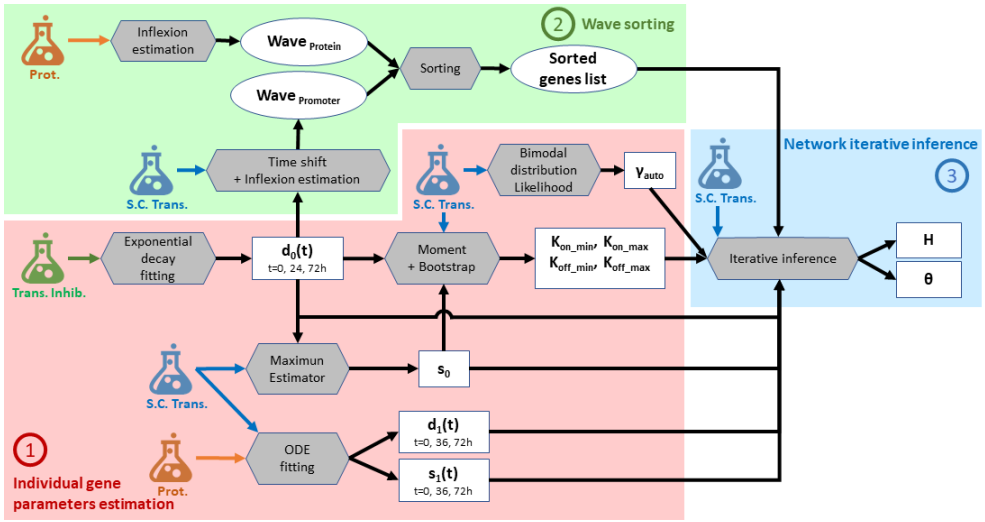


Fig 8. Parameters estimation workflow. Schematic view of WASABI workflow with 3 main steps: (1) individual gene parameters estimation (red zone), (2) waves sorting (green zone) and (3) network iterative interaction inference (blue zone). Wave concept is introduced in "Result" section. Model parameters (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 $d_0(t)$.

$$k_{\text{on\_max\_lim\_min}} = \frac{\max_t(\text{median}_n(k_{\text{on},t,n})) - 0.1 \times k_{\text{on\_min}}}{0.9}$$
$$k_{\text{on\_max\_lim\_max}} = \frac{\max_t(\text{median}_n(k_{\text{on},t,n})) - 0.9 \times k_{\text{on\_min}}}{0.1} \quad (6)$$

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.

$$k_{\text{off\_max\_lim\_min}} = \frac{\max_t(\text{median}_n(k_{\text{off},t,n})) - 0.1 \times k_{\text{off\_min}}}{0.9}$$
$$k_{\text{off\_max\_lim\_max}} = \frac{\max_t(\text{median}_n(k_{\text{off},t,n})) - 0.9 \times k_{\text{off\_min}}}{0.1} \quad (7)$$

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 $+30$h, 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).

Declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Availability of data and material

Single-cell transcriptomic data are available from [37]. Proteomic data are available from [39]. *In silico* generated data are available at `https://osf.io/gkedt/`.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

AB, UH, PAG and OG designed the study. AB performed the theoretical derivations, implemented the algorithms and conceived/analyzed the *in silico* study. UH implemented the algorithm for auto-positive feedback exponent estimation. AR, SG and AG participated in data generation. OG secured the funding. AB drafted the paper. UH, AR, AG, SG, PAG and OG revised the paper. All authors read and approved the final manuscript.

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.

Acknowledgements

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.

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. doi:10.1056/NEJMms1706744.

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;545:432. doi:10.1038/nature22370.

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;545:439. doi:10.1038/nature22326.

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. doi:10.1186/gm340.

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;48:55–65. doi:S0010-4825(14)00042-0 [pii] 10.1016/j.compbiomed.2014.02.011.

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. doi:10.1186/1471-2105-7-249.

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. doi:https://doi.org/10.1016/j.ymeth.2012.10.012.

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. doi:10.1093/bioinformatics/btl364.

14. Polynikins A, Hogan SJ, Bernardo M. Comparing different ODE Modelling approaches forgene regulatory networks. JTheorBiol. 2009;261:511–530.

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, Vento-Tormo R, Teichmann S. Exponential scaling of single-cell RNAseq in the last decade. BioRxiv preprint. 2017;.

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;doi:10.1093/bfgp/elx046.

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. doi:10.1016/j.tig.2013.11.002.

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. doi:10.1002/bies.201500124.

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. doi:10.1016/j.molcel.2016.05.023.

22. Cannoodt R, Saelens W, Saeys Y. Computational methods for trajectory inference from single-cell transcriptomics. Eur J Immunol. 2016;46(11):2496–2506. doi:10.1002/eji.201646347.

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. doi:10.1093/bioinformatics/btu777.

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. doi:10.1186/s12859-016-1235-y.

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;33(3):269–76. doi:10.1038/nbt.3154.

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. doi:10.1186/s12859-016-1109-3.

27. Cordero P, Stuart JM. In: Tracing co-regulatory network dynamics in noisy, single-cell transcriptome trajectories. World scientific; 2016. 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;doi:10.1093/bioinformatics/btx605.

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. doi:10.1093/bioinformatics/btx194.

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. doi:10.1093/bioinformatics/btv257.

31. Huang S. Non-genetic heterogeneity of cells in development: more than just noise. Development. 2009;136(23):3853–62. doi:136/23/3853 [pii] 10.1242/dev.035139.

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. doi:10.1016/j.cels.2015.08.001.

33. Munsky B, Trinh B, Khammash M. Listening to the noise: random fluctuations reveal gene network parameters. Mol Syst Biol. 2009;5:318. doi:msb200975 [pii] 10.1038/msb.2009.75.

34. Moris N, Pina C, Arias AM. Transition states and cell fate decisions in epigenetic landscapes. Nat Rev Genet. 2016;17(11):693–703. doi:10.1038/nrg.2016.98.

35. Papili Gao N, Ud-Dean MSM, Gandrillon O, Gunawan R. SINCERITIES: Inferring gene regulatory networks from time-stamped single cell transcriptional expression profiles. 2016;doi:10.1101/089110.

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. doi:10.1371/journal.pbio.1002585.

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. 2018;doi:10.1101/289728.

40. Liu Z, Tjian R. Visualizing transcription factor dynamics in living cells. J Cell Biol. 2018;doi:10.1083/jcb.201710038.

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. doi:10.1073/pnas.0704167104.

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. doi:10.1074/mcp.O113.034181.

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. doi:10.1038/nrm.2016.139.

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. doi:10.1016/j.ymeth.2017.04.008.

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. doi:10.1371/journal.pcbi.1006000.

48. Ud-Dean SM, Gunawan R. Optimal design of gene knockout experiments for gene regulatory network inference. Bioinformatics. 2016;32(6):875–83. doi:10.1093/bioinformatics/btv672.

49. Kreutz C, Timmer J. Systems biology: experimental design. FEBS J. 2009;276(4):923–42. doi:10.1111/j.1742-4658.2008.06843.x.

50. Semrau S, Goldmann J, Soumillon M, Mikkelsen TS, Jaenisch R, van Oudenaarden A. Lineage commitment revealed by single-cell transcriptomics of differentiating embryonic stem cells. 2016;doi:10.1101/068288.

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. doi:10.7554/eLife.20487.

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. doi:10.1016/j.molcel.2013.01.004.

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. doi:10.1371/journal.pone.0145180.

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.

– Supplementary information –

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

Contents

1 *in vitro* genes parameters and waves estimation

1.1 Table and figures of gene parameters and wave times estimation

All following files are available at `https://osf.io/gkedt/`. Table *in_vitro_gene_parameters_estimation.csv* provides all genes parameters and wave times. Files *Waves_invitro_1_wave_per_gene.pdf* and *Waves_invitro_2_waves_per_gene.pdf* illustrate wave time estimation respectively in case of one wave per gene or 2 waves per gene. File *Protein_fitting.pdf* illustrates protein fitting for s_1 and d_1 parameter estimation.

1.2 Protein parameters correlation

For the 25 genes that were neither detected in our proteomic data or in literature [1], 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 [1] proposed an exponential correlation between s_1 (translation rate) and mean protein level that is confirmed by Fig 1:

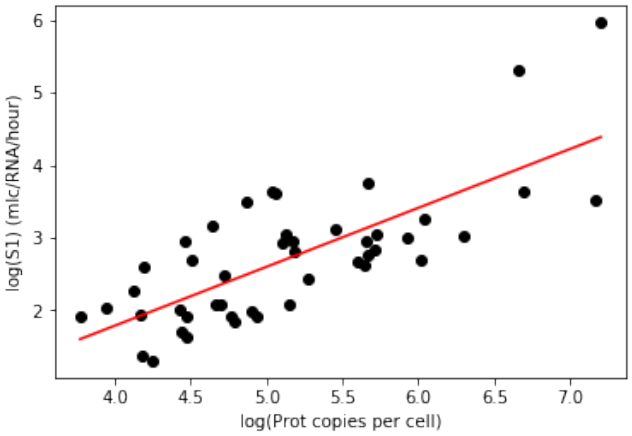


Fig 1. Correlation between s_1 and protein level. Exponential correlation between estimated s_1 and mean protein level. We consider the relation $s_1 = 10^{-1.47} * P^{0.81}$. Linear regression was performed with Python scipy.stats.linregress() function from Scipy package: $r^2 = 0.55$, slope=0.81, intercept=-1.47, p=$2.97 * 10^{-9}$.

1.3 Auto-positive feedback coefficient estimation

Distribution of estimated auto-positive feedback coefficient (Fig 2) from *in vitro* data clearly distinguish 2 groups of genes. One group of 11 genes with a very low coefficient lower than 0.5, and another important group of 79 genes with coefficients greater than 0.5. This result is consistent with assumption of non-autoactivated genes and other influenced by a positive loop. More over, variability of non-null coefficients ranging from 0.5 to 2.25 could carefully be interpreted as presence of strong direct self-activation and weaker positive feedbacks. This last interpretation should be validated by additional work.

Auto-positive feedback coefficients were also estimated from 20 *in silico* GRN embedding autoacivated genes (Fig3). Only genes with auto-positive feedback that are

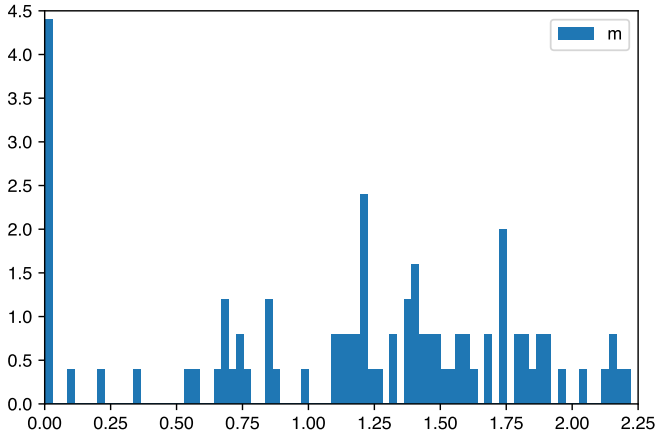


Fig 2. *in vitro* auto-positive feedback coefficient estimation. Interaction auto-positive feedback coefficient parameters estimated from time course single cell RNA distribution. This coefficient corresponds to exponent parameter of Hill like interaction function between gene protein against its own promoter parameters. Null value corresponds to absence of positive feedback loop.

activated, and not inhibited, during simulation have an estimated auto-positive feedback coefficient greater than most of other genes. Remarkably, we observe a threshold around 0.45, like for *in vitro* distribution (Fig2). This similitude gives credit to representativeness of our inslico GRN and comfort our choice to set auto-positive feedback detection threshold to 0.45. However, *in vitro* auto-positive feedback coefficients range to 2 while *in silico* ones are limited to 1, suggesting that biological auto-positive feedback are stronger in intensity compare to our model. But this difference has no impact on the definition of auto-positive feedback detection threshold.

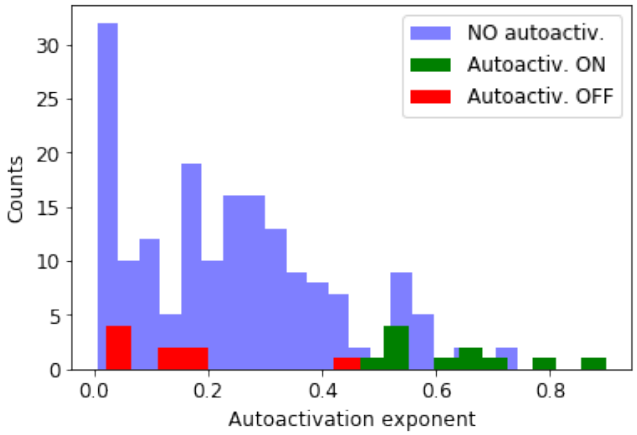


Fig 3. *in silico* auto-positive feedback coefficient estimation. Auto-positive feedback coefficients are estimated from *in silico* single cell data. Genes with an auto-positive feedback that are activated during simulation are presented in green, inhibited are presented in red. Genes without auto-positive feedback are presented in blue.

2 *In silico* benchmarking

2.1 Wave time difference in case of auto-positive feedback

To estimate the acceptable range for wave time difference in case of autoactivated target gene, we reuse the 20 *in silico* GRNs previously used for auto-positive feedback coefficient estimation. For each interactions of these 20 *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 considering regulator gene autoactivation status. Distribution of wave times differences is provided in following Fig 4. Acceptable range for wave times difference in case of auto-activation is set to $[-30h, 50h]$.

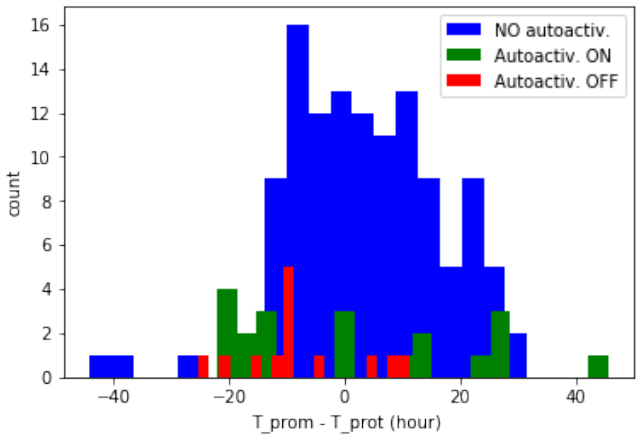


Fig 4. Distribution of wave time difference for auto-positive feedback genes

2.2 *In silico* GRN definition

For *in silico* validation we define 3 GRNs to be inferred which topology is given in Fig 5. Gene's parameters are given in table 1. Interaction parameters are given in table 2.

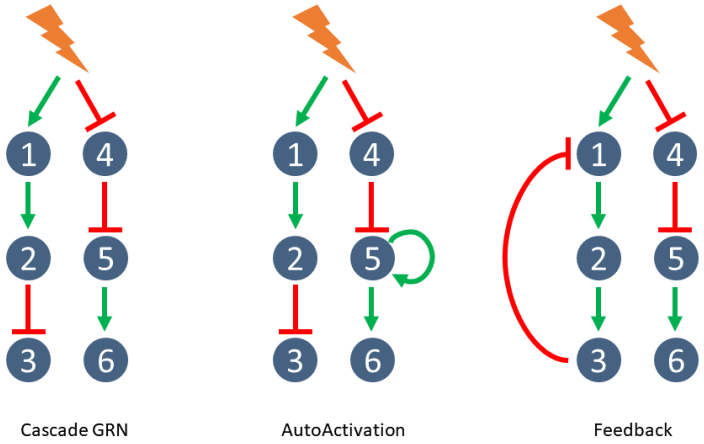


Fig 5. In-silico GRN 3 GRN were designed with different structure pattern to validate WASABI inference

Table 1. *in silico* GRN gene parameters.

parameter	value	unit
d_0	0.173	h^{-1}
s_0	100	h^{-1}
d_1	0.046	h^{-1}
s_1	10000	h^{-1}
$k_{\text{on\_min}}$	0.001	h^{-1}
$k_{\text{off\_min}}$	0.3	h^{-1}
$k_{\text{on\_max}}$	0.1	h^{-1}
$k_{\text{off\_max}}$	2	h^{-1}
dt	0.5	h

Table 2. in silico GRN interaction parameters.

GRN	Regulator	Target	Protein threshold	Efficiency
Cascade	Stim 1	gene 1	0.01	4
	Stim 1	gene 4	0.01	-4
	gene 1	gene 2	6	4
	gene 2	gene 3	5	-4
	gene 4	gene 5	10	-3
	gene 5	gene 6	10	4
Auto-positive feedback	Stim 1	gene 1	0.01	4
	Stim 1	gene 4	0.01	-4
	gene 1	gene 2	6	4
	gene 2	gene 3	2	-4
	gene 4	gene 5	10	-3
	gene 5	gene 6	10	4
	gene 5	gene 5	10	4
Feedback	Stim 1	gene 1	0.01	2
	Stim 1	gene 4	0.01	-4
	gene 1	gene 2	6	4
	gene 2	gene 3	5	4
	gene 4	gene 5	10	-3
	gene 5	gene 6	10	4
	gene 3	gene 1	3.5	-4

2.3 *In silico* experimental data

For the 3 *in silico* network we generate experimental data for time points [0,2,4,8,24,33,48,72,100] hours after continuous step stimulation. Data are available at `https://osf.io/gkedt/`. We simulate 200 cells for single-cell data (RNA counts). The mean of 500 cells gives bulk value for RNA counts and protein concentration (μM).

2.4 *In silico* wave times

For the 3 *in silico* GRNs, wave times for promoter and protein are estimated from simulated bulk data. Wave times are given in hours.

Table 3. *In silico* estimated wave times:ND = Not Detected

GRN	Gene	$W_{promotor}$	$W_{protein}$
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
Auto-positive feedback	1	3.67	16.19
	4	4.07	11.76
	2	18.20	35.02
	3	28.50	33.46
	5	38.40	54.87
	6	52.25	66.69
Feedback	1	-0.90	16.93
	4	-0.71	15.38
	2	12.47	86.84
	5	15.92	40.31
	6	33.00	53.97
	3	37.60	52.75
	1	55.50	ND
	2	65.49	86.84

2.5 *In silico* inference

2.5.1 Definition of inference *Quality*

We note *GRN quality* the inference quality metric that quantifies proportion of true interactions conserved in the candidate network compared to true network. A 100% corresponds to the true GRN. To compute *GRN quality* for a GRN candidate, we first compute for each of its genes a *sub-network quality*, the sub-network corresponds to all paths connecting stimulus to the gene. Then, we compute *GRN quality* as the mean value of all *sub-network qualities*.

sub-network quality is computed for a gene as follow: we estimate the number of intermediaries genes between gene and stimulus in both candidate and true sub-networks. If the numbers of intermediaries is different, *sub-network quality* is null. Else, *sub-network quality* corresponds to the ratio of (i) counts of common interactions between candidate and true sub-networks, and (ii) maximum between candidate and true sub-network sizes (interaction counts).

2.5.2 Cascade GRN

WASABI is run to infer cascade *in silico* network. Interaction consensus matrix Fig 6 is generated for each network candidate with a fit distance lower than 15. Each square in the matrix represents either the absence of any interaction, in dark blue, 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. Sign of frequency indicates activation (positive) or inhibition (negative). Green and red circles respectively correspond to true network activations and inhibitions.

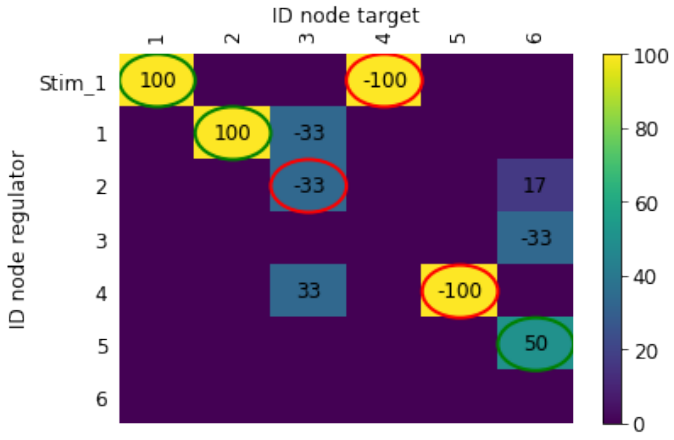


Fig 6. Cascade network consensus interaction matrix.

2.5.3 Auto-positive feedback

Genes auto-positive feedback coefficient are estimated from *in silico* single cell data. According to threshold set to 0.45, only gene 5 of autoactivated network presents an auto-positive feedback. Table 4 gives estimated auto-positive feedback coefficient for all genes of autoactivated network.

Table 4. AutoActivation coefficient estimation.

Gene	Autoactivation coefficient
1	0.19
2	0.21
3	0.067
4	0.14
5	0.65
6	0.28

WASABI is run to infer autoactivated *in silico* network. Fit distance distribution Fig 7 is represented for true GRN (green) and candidates (blue). 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

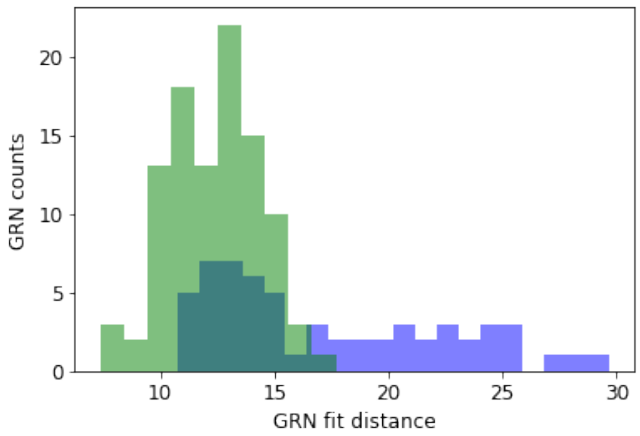


Fig 7. Auto-positive feedback: Fit Distance for true GRN and candidates. Reexpliquer le graph

Interaction consensus matrix Fig 8 is generated for each network candidate with a fit distance lower than 17. See cascade network consensus matrix for figure description.

Fig 8. Autoactivated network consensus interaction matrix.

2.5.4 Feedback GRN

WASABI is run to infer negative feedback *in silico* network. Fit distance distribution Fig 9 is represented for true GRN (green) and candidates (blue). 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

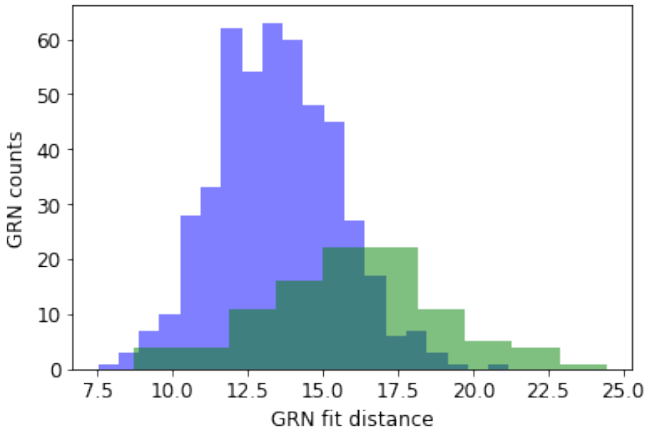


Fig 9. Feedback: Fit Distance for true GRN and candidates.

Interaction consensus matrix Fig 10 is generated for all network candidates. See cascade network consensus matrix for figure description.

Fig 10. Negative feedback consensus interaction matrix.

3 *In vitro* GRN candidates fit distance distribution

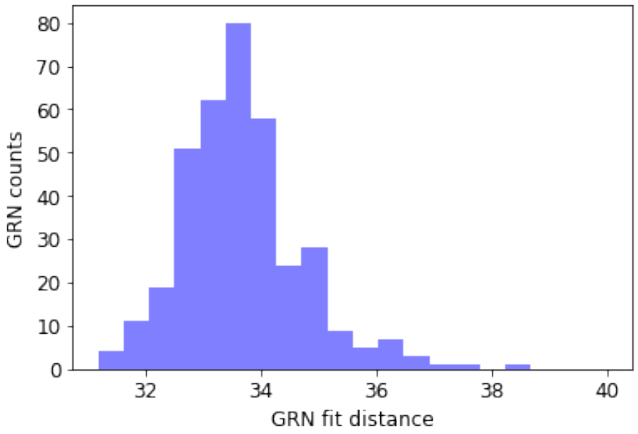


Fig 11. *In vitro* GRN candidates fit distance distribution 364 GRN candidates (excluding outliers) were generated from WASABI application to *in vitro* data.

References

1. 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.

2.4 Article 4 : A Cloud-aware autonomous workflow engine and its application to Gene Regulatory Networks inference

La stratégie de WASABI repose sur la parallélisation de l'inférence, ses performances dépendent donc de la puissance de calcul utilisée. Les solutions de type Cloud offrent une grande puissance de calcul, mais la gestion du flux des tâches pour des applications scientifiques comme WASABI est une problématique complexe étudiée par l'équipe AVALON de l'ENS de Lyon. Nous avons collaboré dans cette étude pour appliquer et tester leur solution d'optimisation de calcul parallèle en Cloud sur une simulation de flux de tâches généré par WASABI.

2.4.1 Principaux résultats de l'article 4

Le flux de tâche de WASABI a été testé et simulé avec différents types d'allocation des tâches, une statique et l'autre automatique avec l'algorithme développé par l'équipe AVALON. Le gain principal vient de la réduction des coûts (tableau 2 de l'article 4). Avec une gestion statique sur un Cloud de Type Amazon le coût est d'un peu moins de 2000$ pour une inférence de WASABI, ce qui est non négligeable. La gestion automatique permet de réduire fortement ce coût de 45% pour le même travail.

2.4.2 Principales conclusions de l'article 4

Ces travaux montrent d'une part qu'il est possible de réduire fortement le coût de déploiement d'un algorithme tel que WASABI sur des plateformes de calcul commerciales, et d'autre part que ce déploiement est adaptable et peut être géré automatiquement sur n'importe quelle plateforme pour faciliter leur utilisation. Ces résultats confirment que le passage à une échelle plus industrielle de WASABI est possible.

2.4.3 Article 4

In Proceedings of the 8th International Conference on Cloud Computing and Services Science - Volume 1 : CLOSER, ISBN 978-989-758-295-0, pages 509-516. DOI : 10.5220/0006772805090516

URL : https ://doi.org/10.1186/s12918-017-0487-0

A Cloud-aware Autonomous Workflow Engine and Its Application to Gene Regulatory Networks Inference

Arnaud Bonnaffoux[1,2], Eddy Caron[3], Hadrien Croubois[3] and Olivier Gandrillon[1]
[1]*Univ. Lyon, ENS de Lyon, Univ. Claude Bernard, CNRS UMR 5239, INSERM U1210, Lyon, France*
[2]*Cosmo Tech, Lyon, France*
[3]*Univ. Lyon, ENS de Lyon, Inria, CNRS, Université Claude-Bernard Lyon 1, Lyon, France*

Keywords: Auto-scaling, Resource Management, Workflow, Cloud, Scientific Applications, HPC.

Abstract: With the recent development of commercial Cloud offers, Cloud solutions are today the obvious solution for many computing use-cases. However, high performance scientific computing is still among the few domains where Cloud still raises more issues than it solves. Notably, combining the workflow representation of complex scientific applications with the dynamic allocation of resources in a Cloud environment is still a major challenge. In the meantime, users with monolithic applications are facing challenges when trying to move from classical HPC hardware to elastic platforms. In this paper, we present the structure of an autonomous workflow manager dedicated to IaaS-based Clouds (Infrastructure as a Service) with DaaS storage services (Data as a Service). The solution proposed in this paper fully handles the execution of multiple workflows on a dynamically allocated shared platform. As a proof of concept we validate our solution through a biologic application with the WASABI workflow.

1 INTRODUCTION

Scientists in fields like biology and physics tend to rely more and more on High Performance Computing (HPC) resources both to perform large scale simulations and to analysis the huge amount of data produced by said simulations as well as other experiments. For example, new generation DNA sequencers can now produce a large amount of data, in the terabyte range, at a very low cost. Analyzing these required the development of computing tools for large scale sequence alignment, which relies upon HPC resources (Das et al., 2017; Yu et al., 2017). Similarly, the reconstruction of Gene Regulatory Networks (GRNs) from high-throughput experimental data comes with a high computational cost, leading to the development of parallel algorithms (Xiao et al., 2015; Zheng et al., 2016; Lee et al., 2014).

However, accessibility to these HPC resources is limited and Cloud-based platforms have emerged as good solution for people with these use-cases that might not have access to large computing infrastructures. Using the virtual resources offered by Cloud providers, anyone can build its own computing platform without having to bear the initial investment cost and the necessary maintenance that comes with owning the hardware. However, while HPC applications are moving toward the Cloud, the deployment mechanisms used are generally trying to replicate the existing paradigm rather than using the full elasticity the Cloud has to offer.

The approach most commonly used is to deploy Cloud instances such to have a platform similar to what users are familiar with, and use the batch scheduling mechanisms they are familiar with. While this approach requires minimal changes, the tools used were designed for a fixed platform, and do not benefit from the dynamicity of Cloud solutions. We, on the other hand, believe that using the dynamicity of Cloud infrastructures to modify the platform deployment in real time can help users achieve better performances at a lower cost. Managing such deployment is a complex task, which requires constant awareness of the platform and of the workload. In order to achieve that, we need the (re)deployment mechanisms to work autonomously. Rather than asking the user to dive into the details of the platform deployment, we have to build a solution that releases them from all interactions with this deployment process.

Unlike task placement, which is a well-studied issue, little work deals with the automation of Cloud platform deployment.

Bonnaffoux, A., Caron, E., Croubois, H. and Gandrillon, O.
A Cloud-aware Autonomous Workflow Engine and Its Application to Gene Regulatory Networks Inference.
In *Proceedings of the 8th International Conference on Cloud Computing and Services Science (CLOSER 2018)*, pages 509-516
ISBN: 978-989-758-295-0

Our goal in this paper is to evaluate the efficiency of our framework, which contains mechanisms that automate the deployment of Cloud ressources into a self adapting, shared, computing platform, as well as scheduling scientific applications on top of it. As a proof of concept we validate our solution through a biologic application with the WASABI workflow. We provide a tool that contribute to the convergence of HPC applications and Cloud resources, thus providing easy access to HPC resources to all users.

2 RELATED WORK

Many scientific and industrial applications from various disciplines are structured as workflows (Bharathi et al., 2008). A workflow can be seen as a structured set of operations which, given an input data set, produce the expected result. For a long time, the development of complex middleware with workflow engine (Couvares et al., 2007; Deelman et al., 2005; Caron et al., 2010) automated workflow management. Infrastructure as a Service (IaaS) Clouds raised a lot of interest recently thanks to an elastic resource allocation and pay-as-you-go billing model. A Cloud user can adapt the execution environment to the needs of their application on a virtually infinite supply of resources. While the elasticity provided by IaaS Clouds gives way to more dynamic application models, it also raises new issues from a scheduling point of view. An execution now corresponds to a certain budget, which imposes certain constraints on the scheduling process.

Solutions to dynamically-scaled Cloud computing instances exist. For example in (Mao et al., 2010) the solution is based on deadline and budget information. In (Kailasam et al., 2010) the solution deals with Cloud bursting. Another autonomous auto-scaling controller, which maintains the optimal number of resources and responds efficiently to workload variations based on the stream of measurements from the system, is introduced in (Londoño-Peláez and Florez-Samur, 2013). This paper shows the benefits of an auto-scaling solution for Cloud deployments. However, these solutions do not handle workflow applications. Likewise in (Nikravesh et al., 2015), authors gave a suitable prediction technique based on the performance pattern, which led to more accurate prediction results. Unfortunately, all these papers fail to consider the delays resulting from communications between the different tasks of a workflow.

In (Mao and Humphrey, 2013), authors show the benefit of auto-scaling to deal with unpredicted workflow jobs. They also show that scheduling-first and scaling-first algorithms have different advantages over each other within different budget ranges. The auto-scaling mechanism is introduced as a promising research direction for future work.

Nevertheless, some solutions provide an autonomous workflow engine. In (Heinis et al., 2005), altering the cluster configuration helps the authors build an autonomous controller that responds to workload variations. Authors introduce a mechanism of self-healing that reacts to changes in the cluster configuration. This solution shows some benefits for cluster architecture but does not work well for Cloud architecture. Workflow engines for Cloud environments dealing with dynamic scalable runtimes are given in (Pandey et al., 2012).

A comparative evaluation of several auto-scaling algorithms is given in (Ilyushkin et al., 2017). However, the policies discussed in the review solely focus on auto-scaling, thus missing on issues like data locality and tasks clustering. Our approach is different in that it considers a more complex issue where workflow optimizations can induce cycles in their clustered representation. This representation ask for more complex task placement and demand analysis mechanisms. Last but not least, our approach differs in that parts of the resource manager and of the scheduler are decentralized and rely on the nodes to take decisions by themselves in order to improve scalability.

3 INFRASTRUCTURE

Our objective in this paper is to describe and evaluate the architecture of a middleware that uses resources from Cloud providers to build a computing infrastructure for the execution of scientific workflows.

3.1 IaaS Cloud Platforms

Among the many offers Cloud providers propose, IaaS (Infrastructure as a Service) is arguably one of the most versatile. Through the use of virtualization technologies, it allows anyone to get access to remote resources and use them just as if they owned their own server. These resources are seen as virtual machines and they can run any system the user needs. This enables anyone to build their own computing infrastructure without the initial investment cost or the burden of maintenance that comes with owning hardware.

The main advantage of Cloud solutions, which explains their development over the past few years, is the versatility and dynamicity of these solutions. Not only can a user deploy a custom platform in just minutes, but the deployment can be modified to match any change in the workload. As such, users only have to

pay for what is really needed, and not bear the cost of platform when not in use.

3.2 The Allocation Problem

It is easy to deploy a large platform using current Cloud technologies. However, knowing how many resources are really needed is a completely different problem. There might be many options, each one resulting in different performances and costs. While cost per unit of time is easy to compute, estimating performances and tasks completion time is extremely difficult. Consequently, the total deployment cost can also be hard to predict as it depends on completion time.

A common practice is to distribute the budget along a specific duration and get as many resources as one can afford during this period. This simple approach, which tries to maximize performances within a given cost constraint, has major drawbacks. Not only is the user committing all his budget, but there are no warranties to have enough resources during peak hours, and resources are very likely to be wasted during off-peak hours.

A more elegant and efficient solution would be to modify the deployment in real time so that it matches the needs. Estimating the needs is a very complex task, and adjustment should be performed 24/7, which would require too many actions to be realistically performed by a human. For it to be efficient, we need this process to be fully autonomous and not require any human input.

3.3 An Autonomic Solution

In order to automate the deployment of the platform, we have to design a control loop that could drive the allocation mechanisms. According to the MAPE-K (Kephart and Chess, 2003) model, such a loop requires 4 features: (1) Monitoring the platform, both in terms of platform status and workload; (2) Analyzing the needs in term of platform allocation; (3) Planning actions to modify platform allocation towards what is required; (4) Executing the plan.

Such design is already part of common schedulers and is used to control the placement of the interdependent tasks in a workflow. However, unlike dependency control, the issue of platform deployment control is complex and still unsolved. This is this issue that our approach (Figure 1) tries to solve in the context of homogeneous IaaS Clouds.

In addition to the traditional mechanisms used to control the task flow (scheduling loop), we added features to the existing agents, as well as new agents,

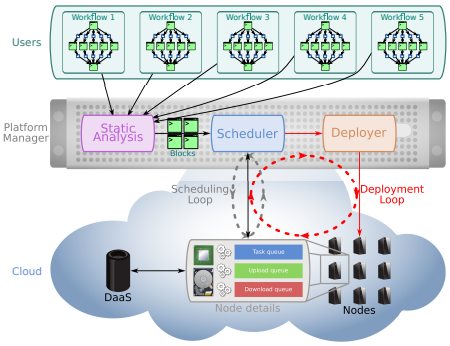


Figure 1: Outlines of our middleware with autonomous platform deployment capabilities.

to implement a deployment control loop. More precisely, we built these such that two distinct mechanisms work in coordination to achieve the automation of both up-scaling and down-scaling.

Data Locality. Data locality is a major concern in the scheduling of workflows. While parallelization is expected to reduce the completion time of workflows, data transfer can produce undesirable side effect that reduces the overall performances.

In our previous work (Caron and Croubois, 2017) we discussed the possibility to solve this issue using an off-line scheduling mechanism. The static analysis step described in this paper left us with blocks of tasks that already handle the issue of data locality and can therefore be scheduled without requiring the scheduler to solve this optimization issues at runtime.

Down-scaling Control. The down-scaling is decentralized and controlled by the nodes themselves. Each node has its own work queue, and requests work from the scheduler when it has free resources in term of CPU (ability to start new work) or Down-link (ability to prefetch data). When a node requests work, the scheduler is to provide it with a block that can be executed on this node without breaking the deadline constraints. However, if the platform is oversized, the nodes will end up executing all the tasks up to the point where the scheduler cannot provide work to the requesting node. In addition to requesting work, when a node work queue is empty, the node starts a suicide timer. Once this timer is initiated, and if no work has been received by then, the node will automatically deallocate itself before the beginning of the next billing hour.

This mechanism helps reduce the dimension of oversized platforms during off-peak hours thus enabling a more efficient platform deployment that decreases infrastructure cost.

Up-scaling Control. Unlike the down-scaling, the up-scaling mechanism is not decentralized, and requires a global knowledge of the platform. It relies on a new agent, the *deployer*. When the workload changes, for example with the addition of new jobs by the users, the scheduler will inform the deployer of these changes, and provide it with a description of the work queues and nodes status. This monitoring of the platform means that the deployer is able to use simple list scheduling algorithms to simulate jobs placement, analyze it, and plan for the deployment of new nodes if required.

If new nodes have to be deployed, the deployer will do so according to the last computed deployment schedule. This schedule can be overridden by new, updated, runs of the analysis and planning steps. This control mechanism ensures new nodes are deployed when required, such that QoS is achieved (deadlines are meet).

4 GENE REGULATORY NETWORKS INFERENCE

4.1 WASABI

Gene Regulatory Networks (GRN) play an important role in many biological processes, such as cell differentiation, and their identification has raised great expectations for understanding cell behaviors. Many computational GRN inference approaches are based on bulk expression data, and they face common issues such as data scarcity, high dimensionality or population blurring (Chai et al., 2014). We believe that recent high-throughput single cell expression data (see (Pina et al., 2012)) acquired in time-series will allow to overcome these issues and give access to causality, instead of "simple" correlations, to dissect gene interactions. Causality is very important for mechanistic model inference and biological relevance because it enables the emergence of cellular decision-making. Emergent properties of a mechanistic model of a GRN should then match with multi-scale (molecular/cellular) and multi-level (single cell/population) observations.

The WASABI (WAves Analysis Based Inference) framework is based upon the idea of an iterative inference method. This will allow to adopt a divide-and-conquer type of approach, where the complexity of the problem is broken down to a "one gene at a time" much simpler problem, which can be parallelized.

The whole process can be decomposed along the following steps:

1. Gene Ordering. We order all the 94 genes from time-series single cell gene expression data from chicken erythrocyte progenitors acquired during their differentiation process (Richard et al., 2016). These data have been analyzed using a stochastic mechanistic model of gene expression, the Random Telegraph model (Peccoud and Ycart, 1995).
2. Iterative inference. For each step of the inference process a new gene is added to a set of GRN candidates inferred in the previous iterations. It creates new extended GRN candidates to be assessed, though all possible combinations with the previously conserved networks. For each new network, its behavior will be simulated using a recently described mathematical formalism (Herbach et al., 2017) and the resulting gene expression values will be compared to the experimental one. Should the fit between the two genes expression be acceptable, the networks will be kept and carried to the next step. If, on the contrary, the network should be too different, then it will be pruned and will not participate in future attempts.

At the beginning of the process, one would expect (and an initial assessment confirmed) that the number of suitable networks will sharply increase. In this growth phase, an efficient parallelization will be of essence. In a second phase, one expects that most of the "bad" networks will have accumulated so many errors that they will diverge from the experimental reality, and it will in the end result in the generation of a manageable amount of networks.

The use of Design of Experiment approaches (Kreutz and Timmer, 2009) should finally help us to get to the most probable network.

4.2 WASABI Workflow Description

WASABI, as an application, is composed of many steps, each one divided into many instances of the same elements. The control flow, which links these many elements into the complete application, can express this iterative structure through a DAG of successive fork-join patterns. The WASABI workflow contains 9 fork-join steps of width 5, 6, 36, 252, 1000, 1000, 1000, 1000, 1000. This adds up to a total 5309 tasks (including synchronization tasks) and 10598 control flow dependencies. Total runtime for all these tasks is 4250.1 hours.

All these informations, as well as details about individual tasks, such as runtime) were extracted from traces of the previous runs. This allowed us to build a descriptions of WASABI as a workflow. While our middleware is not based on the Pegasus engine, for the sake of clarity, our inputs file are compliant with

the well-known Pegasus workflow syntax. It is this workflow that we will be using to compare our deployment solution to existing approaches.

5 EVALUATION

5.1 Comparison to Fixed Deployment for a Single Workflow Execution

In order to estimate the efficiency of our deployment, we will compare it to a more traditional "fixed deployment" method. In a fixed deployment, the user books a given amount of resources which are then used by a batch scheduler. Whenever a task is ready to run and a node is available, the task will be placed on the node to be computed. It is clear that, with this approach, the more resources there are, the faster the workflows will be executed, up to the point where the sequential dimension of the workflow prevents the user from achieving any more parallelism. Still, having more resources also means that more cpu time is wasted during the synchronization parts of the workload. Once all computation is done, the user has to release the resources.

Assessing total cost and runtime of a workflow for a specific deployment is a hazardous operation which requires trial and error. Simulation can help with this process, but it requires knowledge of the platform specifications and time, which users non familiar with computer science might not have. What our framework offers is a platform where users can specify their workflows and the required wall time for each of them. The platform is then automatically deployed in order to meet the expected QoS while achieving the lowest cost possible.

The results in this section have be achieved through simulation. Details about the WASABI workflow were extracted from traces of runs on existing HPC structures (IN2P3). Estimation of the deployment cost for those simulations were obtained assuming their deployment on Amazon EC2 instances of similar performance. The down-scaling mechanism was tunned to match Amazon EC2 billing policy.

Our first results show platforms performance and cost for deploying a single WASABI workflow. This workflow contains 5309 tasks for a total of 4250 core-hours. Ideally, we would like to pay only for these 4250 hours of computation, but the billing policy of Cloud providers is such that we will also be charged for some unused time when we cannot perfectly use the hours allocated to each node. For example, a node used to compute a task that lasts 1 hour and 48 minutes will be billed for 2 hours, and we are thus wasting 12 minutes of CPU time. The critical path in this workflow is about 17 hours, meaning that no matter how many resources we have, we will not be able to get results faster than that using the type of node considered here.

For this experiment, we run simulations with fixed platforms of sizes varying from 100 to 400 nodes. This gives us a base line of what current approaches achieve. The results in Table. 1b and Figure 1a show a Pareto front which comes from the conflict between two contradictory objectives: maximizing platform performance and minimizing deployment cost. This confirms the idea that there is not ideal value for the size of a fixed platform. Selecting the size of a fixed platform results in a choice between performance and cost on this non-optimal front.

However, the dynamicity offered by the deployment control loop implemented in our framework leads to better results. By efficiently allocating and deallocating nodes we manage to free ourselves from this Pareto efficiency and we achieve good results in term of deployment cost even when facing tight QoS constraints.

5.2 Multi-tenant/Multi-workflows Deployment: Example of a Small Lab Using WASABI

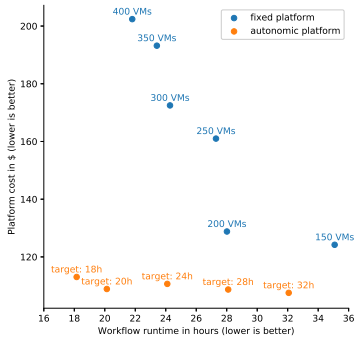
In the previous section, we saw how our approach can efficiently deploy a computing platform to execute a single instance of the WASABI workflow. However, this context still requires quite many human interventions as the platform was dedicated to this workflow. This means that in order to execute their workflow, the user first has to deploy the platform manager. Even worse, in the case of a fixed allocation, the user has to deploy the fixed platform, selecting the number of nodes they want and to be there at the end of the run to shut down the platform.

We believe that the platform should be maintenance-free, and users should be able to submit workflows to an already existing platform manager. This perpetually running tool would be the entry where any user can submit their workflows. The platform would be handled automatically, allocating new nodes when needed, sharing nodes between workflows of different users and shutting nodes down when they are no longer necessary.

Sharing a computing platform today, using the fixed deployment approach and a batch scheduler, is simple but very inefficient. In addition to the waste we could already witness when running a single workflow, we also have to consider all the wasted resources

Table 1: Details for different runs of the WASABI workflow on various platforms. The workflow contains 5309 tasks for a total of 4250.1 core-hours.

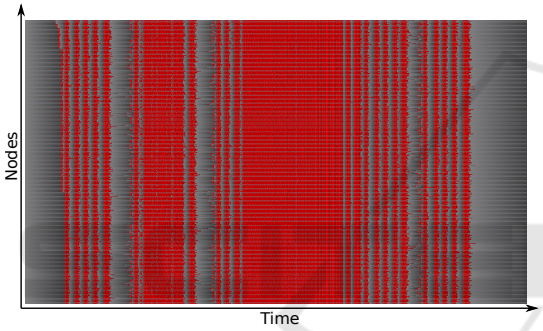
(a) Cost/Makespan front



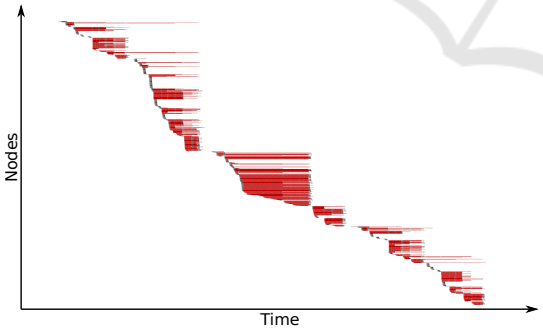
(b) Statistics

Platform	#VMs	Walltime	VM duration	Total usage	Cost	Efficiency
Fixed (100)	100	47:10:51	48h	4800 core-hours	$110.400	88.54%
Fixed (150)	150	35:04:51	36h	5400 core-hours	$124.200	78.71%
Fixed (200)	200	28:00:49	28h	5600 core-hours	$128.800	75.89%
Fixed (250)	250	27:17:42	28h	7000 core-hours	$161.000	60.72%
Fixed (300)	300	24:16:44	25h	7500 core-hours	$172.500	56.67%
Fixed (350)	350	23:24:56	24h	8400 core-hours	$193.200	50.60%
Fixed (400)	400	21:48:05	22h	8800 core-hours	$202.400	48.30%
Autonomous (18h)	851	18:09:14	Variable	4915 core-hours	$113.045	86.47%
Autonomous (20h)	711	20:08:05	Variable	4734 core-hours	$108.882	89.78%
Autonomous (24h)	653	24:06:05	Variable	4811 core-hours	$110.653	88.34%
Autonomous (28h)	676	28:05:46	Variable	4726 core-hours	$108.698	89.93%
Autonomous (32h)	663	32:03:59	Variable	4676 core-hours	$107.548	90.89%

Note: Cost were computed assuming Amazon EC2 *t2.small* instances for an on-demand price of $0.023/hour.



(a) Fixed platform



(b) Autonomous platform

Figure 2: Gantt diagram of VMs during the multi-tenants execution of WASABI workflows. 10 workflows are executed over a one week period, for a total of 42501.0 core-hours of computation. For each VM (line) computation time is shown in red and idle time in grey. Figure 2a shows results for fixed deployment of 500 VMs running for the fun weeks while Figure 2b shows result for an autonomous deployment that spawned 4698 VMs over the week, some of which only ran for a single hour.

which are allocated during off-peak hours.

We simulated such a context at the scale of a small laboratory. In this example, we consider a research team who uses the WASABI application to analyze their results. Over a one week period, our imaginary lab produces data requiring 10 runs of WASABI. Researchers in this team submit the data for analysis when available. As such, the submissions are not distributed evenly throughout the week. In particular, we assume that all submissions will be performed during work days. While no new jobs are submitted during the week-end, computation can still be performed during the week-end. When using the autonomous approach, they ask the results to be available 24 hours after submission.

Statistics for this simulated workload is reported in Table. 2. Gantt diagrams for both deployment methods are shown in Figure 2. Once again, we see that the dynamic allocation of node has major advantages over the previous approach.

The most obvious advantage to our method is cost. Having a platform with 500 *t2.small* instances running for a whole week would cost $1932.00, with an efficiency of 50.60% with this particular workload. For the same work, our approach would reduce deployment cost by 44.57%.

The second advantage to our method is the ease of maintenance and the efficiency of the elastic deployment.

Some might argue that the fixed platform does not have the right size, and using 500 nodes is an empirical choice, and they would be right. However, as discussed previously, choosing which platform size to use is not an obvious choice. While having a 500 nodes platform leads to wasted resources, it is also not enough to ensure that sufficient resources are avail-

Table 2: Platforms statistics for the multi-tenants execution of WASABI workflows. 10 workflows are executed over a 1 week period, for a total of 42501.0 core-hours of computation.

Platform	#VMs	Fastest run	Slowest run	Total VM usage	Cost	Efficiency
Fixed	500	20:12:36	33:58:52	84000 core-hours	$1932.00	50.60%
Autonomous	4698	23:50:03	24:07:12	46563 core-hours	$1070.95	91.28%

Note: Cost were computed assuming Amazon EC2 *t2.small* instances for an on-demand price of $0.023/hour.

able to all users. During rush periods, workflows are fighting for resources. This is visible around the middle of Figure 2a when all nodes are busy. During this period, no resources are wasted but work does not progress as users would expect, causing some workflows to finish with delays. While users would require the workflows to be computed within 24 hours, some took over 33 hours due to the limited number of resources available.

Unlike the fixed platform approach, the autonomous deployment allocates resources when needed which limits waste during off-peak hours and guarantees that the user will have the results they expect with minimal delays. This is visible in Figure 2b. The burst in the workload causes the nodes to be allocated and deallocated in blocks. When many resources are required, the platform manager deploys many nodes. Once the work has been done, nodes start to suffer work shortage and start deallocating. We can see that the burst that caused the fixed platform to saturate is here triggering the allocation of many nodes. These nodes execute tasks from all active workflows and contribute to meeting the QoS users expect. In our example, this elastic allocation process helped limiting delays to at most 7 minutes 12 second.

We also notice that the efficiency achieved by our method in the multi-tenants context is slightly better than in a single workflow context. On a fixed platform, having multiple workflows causes conflicts and decreases the QoS. However, with our elastic deployment manager, having multiple workflows on the same platform is a good thing as the workflows can share nodes to maximize their use and reduce waste. QoS is maintained by the allocation of new resources when required.

Last but not least, the autonomous deployment makes the results availability more predictable. This is more comfortable for users who might require these results for specific deadlines or just to be part of a broader pipeline. With a fixed platform, users are affected by each other's submissions which might lead to frustration and conflicts.

6 CONCLUSION

In this paper, we presented the outlines of a fully autonomous platform manager. We also described the WASABI application, and how it can benefit from this platform management. Comparing this approach to more traditional ones, we showed that not only do we achieve the QoS required by the users through deadlines, but we also achieve substantial savings in deployment cost.

So far this approach is restricted to homogeneous infrastructures, and further work is required to make this solution more versatile. Other optimization objectives, such as minimizing a workflow execution time with a constrained budget, also require some work. Still, our multi-tenants example showed that using our platform to deploy existing scientific workloads can result in cost savings of 40% while avoiding the need for human intervention in the management of the platform.

In conjunction with previous work on the static clustering of workflows for DaaS-based Cloud execution, we now have a comprehensive solution. While the modular nature of this framework means the components could be improved, the current state of this solution is already effective and we hope to apply it soon in the context of an easy-to-use, all-in-one solution for scientific and industrial users.

REFERENCES

Bharathi, S., Chervenak, A., Deelman, E., Mehta, G., Su, M.-H., and Vahi, K. (2008). Characterization of scientific workflows. In *SC'08 Workshop: The 3rd Workshop on Workflows in Support of Large-scale Science (WORKS08) web site*, Austin, TX. ACM/IEEE.

Caron, E. and Croubois, H. (2017). Communication aware task placement for workflow scheduling on daas-based cloud. In *Workshop PDCO 2017. Parallel / Distributed Computing and Optimization*, Orlando, FL. USA. In conjunction with IPDPS 2017, The 31st IEEE International Parallel & Distributed Processing Symposium.

Caron, E., Desprez, F., Glatard, T., Ketan, M., Montagnat, J., and Reimert, D. (2010). Workflow-based comparison of two distributed computing infrastructures. In *Workflows in Support of Large-Scale Science (WORKS10)*, New Orleans. In Conjunction with Supercomputing 10 (SC'10), IEEE. hal-00677820.

Chai, L. E., Loh, S. K., Low, S. T., Mohamad, M. S., Deris, S., and Zakaria, Z. (2014). A review on the computational approaches for gene regulatory network construction. *Comput Biol Med*, 48:55–65.

Couvares, P., Kosar, T., Roy, A., Weber, J., and Wenger, K. (2007). Workflow Management in Condor. In Taylor, I., Deelman, E., Gannon, D., and Shields, M., editors, *Workflows for e-Science*, pages 357–375. Springer.

Das, A. K., Koppa, P. K., Goswami, S., Platania, R., and Park, S. J. (2017). Large-scale parallel genome assembler over cloud computing environment. *J Bioinform Comput Biol*, 15(3):1740003.

Deelman, E., Singh, G., Su, M.-H., Blythe, J., Gil, Y., Kesselman, C., Mehta, G., Vahi, K., Berriman, G. B., Good, J., Laity, A., Jacob, J., and Katz, D. (2005). Pegasus: a Framework for Mapping Complex Scientific Workflows onto Distributed Systems. *Scientific Programming Journal*, 13(3):219–237.

Heinis, T., Pautasso, C., and Alonso, G. (2005). Design and evaluation of an autonomic workflow engine. In *Autonomic Computing, 2005. ICAC 2005. Proceedings. Second International Conference on*, pages 27–38. IEEE.

Herbach, U., Bonnaffoux, A., Espinasse, T., and Gandrillon, O. (2017). Inferring gene regulatory networks from single-cell data: a mechanistic approach. *https://arxiv.org/abs/1705.03407*.

Ilyushkin, A., Ali-Eldin, A., Herbst, N., Papadopoulos, A. V., Ghit, B., Epema, D., and Iosup, A. (2017). An experimental performance evaluation of autoscaling policies for complex workflows. In Binder, W., Cortellessa, V., Koziolek, A., Smirni, E., and Poess, M., editors, *ICPE*, pages 75–86. ACM.

Kailasam, S., Gnanasambandam, N., Dharanipragada, J., and Sharma, N. (2010). Optimizing service level agreements for autonomic cloud bursting schedulers. In *Parallel Processing Workshops (ICPPW), 2010 39th International Conference on*, pages 285–294. IEEE.

Kephart, J. O. and Chess, D. M. (2003). The vision of autonomic computing. *Computer*, 36(1):41–50.

Kreutz, C. and Timmer, J. (2009). Systems biology: experimental design. *FEBS J*, 276(4):923–42.

Lee, W. P., Hsiao, Y. T., and Hwang, W. C. (2014). Designing a parallel evolutionary algorithm for inferring gene networks on the cloud computing environment. *BMC Syst Biol*, 8:5.

Londoño-Peláez, J. M. and Florez-Samur, C. A. (2013). An autonomic auto-scaling controller for cloud based applications. *International Journal of Advanced Computer Science and Applications(IJACSA)*, 4(9).

Mao, M. and Humphrey, M. (2013). Scaling and scheduling to maximize application performance within budget constraints in cloud workflows. In *Parallel & Distributed Processing (IPDPS), 2013 IEEE 27th International Symposium on*, pages 67–78. IEEE.

Mao, M., Li, J., and Humphrey, M. (2010). Cloud auto-scaling with deadline and budget constraints. In *Grid Computing (GRID), 2010 11th IEEE/ACM International Conference on*, pages 41–48. IEEE.

Nikravesh, A. Y., Ajila, S. A., and Lung, C.-H. (2015). Towards an autonomic auto-scaling prediction system for cloud resource provisioning. In Inverardi, P. and Schmerl, B. R., editors, *10th IEEE/ACM International Symposium on Software Engineering for Adaptive and Self-Managing Systems, SEAMS 2015, Florence, Italy, May 18-19, 2015*, pages 35–45. IEEE Computer Society.

Pandey, S., Voorsluys, W., Niu, S., Khandoker, A., and Buyya, R. (2012). An autonomic cloud environment for hosting ecg data analysis services. *Future Generation Computer Systems*, 28(1):147–154.

Peccoud, J. and Ycart, B. (1995). Markovian modelling of gene product synthesis. *Theoretical population biology*, 48:222–234.

Pina, C., Fugazza, C., Tipping, A. J., Brown, J., Soneji, S., Teles, J., Peterson, C., and Enver, T. (2012). Inferring rules of lineage commitment in haematopoiesis. *Nat Cell Biol*, 14(3):287–94.

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., and Gandrillon, O. (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.

Xiao, X., Zhang, W., and Zou, X. (2015). A new asynchronous parallel algorithm for inferring large-scale gene regulatory networks. *PLoS One*, 10(3):e0119294.

Yu, J., Blom, J., Sczyrba, A., and Goesmann, A. (2017). Rapid protein alignment in the cloud: Hamond combines fast diamond alignments with hadoop parallelism. *J Biotechnol*.

Zheng, G., Xu, Y., Zhang, X., Liu, Z. P., Wang, Z., Chen, L., and Zhu, X. G. (2016). Cmip: a software package capable of reconstructing genome-wide regulatory networks using gene expression data. *BMC Bioinformatics*, 17(Suppl 17):535.

3 Discussion et perspectives

3.1 L'utilisation d'approches et de concepts de l'ingénierie repoussent les limites de l'inférence des GRN

Comme je l'ai présenté en introduction, cette thèse au-delà de sa problématique d'inférence des RRG s'inscrit dans un projet personnel de reconversion. L'objectif est d'appliquer les outils, approches et concepts du domaine de l'ingénierie aux problématiques biologiques. Je commencerai donc cette discussion en revenant sur les concepts et résultats qui montrent la pertinence d'une approche d'ingénierie pour l'inférence des RRG.

3.1.1 La causalité trahie par le transitoire

L'utilisation de modèles mécanistes est très largement répandue dans l'industrie pour concevoir les systèmes, les tester ou comprendre des anomalies. Ces modèles requièrent la connaissance des règles de causalité afin de simuler le résultats de scenarii jamais réalisés. Pour cela, il est nécessaire au préalable d'identifier les paramètres de ces modèles à partir de données expérimentales, ce qui montre bien la similitude avec notre problématique. L'étude de la dynamique transitoire pour identifier les liens de causalité est un concept très utilisé dans l'industrie, en particulier dans l'aéronautique. L'idée est de partir d'un état stable, puis de le perturber par un stimulus pour étudier sa réponse transitoire, qui par définition est un état hors-équilibre ou les forces antagonistes ne se sont pas encore équilibrées. On cherche dans le transitoire à identifier des modes et leurs fréquences, ou des déphasages qui trahiraient des liens de causalité. C'est ce type d'approche, que j'ai mis en pratique dans mon expérience passée, que j'ai appliqué à la problématique de l'inférence des RRG. L'équipe SBDM s'intéressait déjà avant mon arrivé à l'évolution dynamique des propriétés RRG, comme l'entropie. Afin de valider si les RRG avait un transitoire au cours de la différenciation que l'on pourrait exploiter pour l'inférence, j'ai demandé très tôt à valider expérimentalement l'observation de "vague d'expression" au cours de la réponse initiale comme cela a été démontré dans l'article 1 (figure 7). Ce premier résultat expérimental a été très important pour moi car il a permis de montrer d'une part que les RRG ont une inertie de plusieurs heures qui permet

l'étude en pratique du transitoire. D'autre part il a montré que l'on pouvait quantifier cette dynamique pour bien distinguer l'ordre de régulation de chaque gène comme je l'ai fait dans l'article 3 en définissant le concept de vague avec les temps de régulation du promoteur et de la protéine d'un gène. Le concept de vague est une des contributions originales de mon travail. Elle permet d'identifier les causalités sur la règle de la temporalité (la cause précède l'effet), plutôt que sur une notion d'indépendance probabiliste.

Pour appliquer mon algorithme WASABI d'inférence de RRG il faut en pratique mesurer l'état initial et final du système et sur-échantillonner temporellement la réponse initiale, ce qui amène à se poser la question de la dynamique du système. Cette question permet de faire le tri dans les processus impliqués et de les étudier séparément selon leur dynamique. C'est le principe de séparation des échelles de temps que l'on a appliqué dans l'article 2 pour la définition du modèle mécaniste des RRG. La notion de causalité doit donc être interprétée avec recul par les biologistes car elle ne se restreint pas à la seule interaction entre un facteur de transcription et sa cible comme beaucoup d'approches le font [78, 80, 81, 82, 103, 104]. Il peut y avoir, et il y a certainement, des acteurs intermédiaires avec des dynamiques beaucoup plus rapides que l'expression génétique, qui devront être identifiés par d'autres expériences spécifiques. La modélisation et les algorithmes d'inférence ne peuvent être la seule solution et doivent être utilisés comme des outils par les biologistes pour guider leurs travaux. Je développerai cette idée plus en détails dans l'étude des perspectives.

3.1.2 Briser le réseau (et la malédiction de la combinatoire) pour le reconstruire fidèlement au fils du temps

Comme nous l'avons vu en introduction, dans la grande famille des algorithmes d'inférence des RRG il y a une diversité d'approches avec leur avantages et inconvénients, mais toutes se heurtent à la malédiction de la combinatoire. Le prix à payer en représentativité biologique pour contourner cette problématique est important, comme la sur-simplification des modèles, la limitation de la taille des réseaux inférés ou les hypothèses arbitraires de rareté des interactions. A mon sens, il faut veiller à ne pas dégrader la représentativité biologique du modèle sous une certaine limite, au-delà de laquelle la problématique d'inférence de RRG devient un pur problème de mathématique déconnecté des enjeux biologiques.

Ainsi, la principale rupture méthodologique de WASABI vient de la volonté de s'attaquer au problème de l'inférence des RRG à l'aide d'un modèle mécaniste biologiquement pertinent, et de l'exploiter par une approche dite "force brute" grâce à la puissance de calcul actuelle. Dans l'aéronautique, des modèles avions mécanistes basés sur des EDO ont été simulés bien avant l'arrivée des ordinateurs, à l'aide de circuits électroniques. Des décennies plus tard, ces modèles ont évolué et sont aujourd'hui utilisés dans des simulateurs de vols ultra-réalistes, ou par les équipes d'ingénierie qui simulent des scénarii sur ordinateurs. On ne cherche pas à simplifier le modèle pour le résoudre à l'aide d'outils mathématiques, sauf dans des cas très particulier, circoncis à des domaines de validité très restreints. Pour les biologistes il est important de bien distinguer l'utilisation des mathématiques d'une part pour formaliser des hypothèses biologiques dans un modèle, et d'autre part pour "résoudre" ce modèle afin de faire des prédictions, de l'inférence ou de l'analyse de données. Bien souvent le modèle mathématique est confié aux mathématiciens qui naturellement cherchent résoudre le problème mathématiquement. C'est pourquoi dans WASABI, je n'utilise que la version du modèle dite de PDMP-couplés, qui a mon sens est un bon compromis entre représentativité biologique et simplicité, que je simule numériquement pour recouper les données expérimentales.

Cependant, l'utilisation de modèle biologique pertinent ne résout en rien la malédiction de la combinatoire. C'est l'exploitation de l'information contenue dans le transitoire, associée à la puissance de calcul actuelle qui permet à WASABI de briser cette malédiction. L'idée d'utiliser des cinétiques pour inférer les RRG n'est pas nouvelle. On a déjà vu dans l'introduction que des cinétiques sur puces ont largement été utilisées [27, 28]. Mais cette information temporelle n'est utilisée dans ces approches que pour fournir plus de puissance statistique pour le recoupement des modèles. Dans les approches statistiques Bayésiennes, le temps peut être utilisé pour recaler les données ou définir une mémoire [40]. Si l'information temporelle est utilisée d'une façon ou d'une autre, elle ne permet pas dans ces approches de s'affranchir du problème de la combinatoire. Comme nous le détaillons dans l'article 3, c'est la définition même de la causalité, qui précède l'effet, qui permet de reconstruire le réseau gène par gène. Il reste une part de combinatoire à chaque itération, mais celle-ci est limitée par les contraintes sur les propriétés dynamiques des gènes. De plus, la parallèlisation du processus d'inférence itératif rend le temps de calcul linéairement dépendant aux nombres

de gènes du réseau, ce qui autorise un passage à l'échelle réaliste pour des réseaux de plus grande taille. C'est certainement la contribution la plus importante de mon travail, qui fait que grâce à WASABI, le problème de l'inférence des RRG n'est plus limité que par des contraintes techniques (capacités de calcul, qualité et quantité des données expérimentales) qui peuvent être déjà surmontées (comme le montre l'article 4) ou le seront dans un avenir proche. Je reviendrai sur ce point dans l'étude des perspectives.

3.1.3 Capitalisation et intégration de données dynamiques multi-échelles

Avec l'arrivée des techniques de donnée à haut-débit la tendance est plutôt à l'analyse d'un seul type de donnée, en particulier les transcriptomes. La revue en introduction des algorithmes d'inférence de RRG montre bien que la très grande majorité ne considèrent que des données de transcriptome. On mesure bien la limite de la pertinence biologique de ces approches qui ne considèrent pas, par exemple, la possibilité d'une régulation post-traductionnelle. Pourtant les moyens de mesure au niveau protéique existent comme nous l'avons montré et appliqué [110]. Une fois encore, un parallèle peut être établi avec l'ingénierie qui a une longue expérience dans l'intégration de données dynamiques hétérogènes. Lors des essais en vol ce sont quelques milliers de variables qui sont enregistrés plusieurs fois par seconde pendant des heures. La quantité de donnée générée est donc équivalente aux expériences biologiques de haut débit. L'analyse de cette montagne de donnée ne peut se faire qu'à l'aide de modèles mécanistes réalistes, qui grâce aux règles d'interactions entre les états du système qu'ils encodent, permettent de faire le lien et l'intégration entre les différents types de mesures. C'est cette démarche que j'ai menée dans WASABI. Dans l'article 3 nous montrons bien comment nous intégrons des données cinétiques transcriptomiques d'inhibition de la transcription en population, des données cinétique transcriptomique en cellule-unique, des données cinétiques de protéomique en population, mais aussi des mesures de la variation du volume moyen des cellules au cours de la différenciation ainsi que des mesures relatives au cycle cellulaire. Toutes ces données sont nécessaires pour estimer au préalable les paramètres propres aux gènes, qui définissent leur dynamique, et enfin pour estimer leur interaction. La figure 8 de l'article 3 montre l'intégration croisée de ces données dynamiques multi-échelles. On notera que des paramètres, comme les taux de dégradations, ne sont pas constants mais

évoluent au cours du temps, ce qui est une innovation par rapport à toutes les approches existantes. C'est ce que l'on observe expérimentalement et que j'ai intégré dans notre modèle. Notamment, le taux de dégradation des protéines de nombreux gènes augmente subitement au cours de la différenciation, laissant sous-entendre une régulation post-traductionnelle forte. Si la résolution mathématique d'un modèle avec des paramètres variables est très peu réalisable, elle ne pose aucun problème supplémentaire à une approche heuristique basée sur la résolution numérique comme nous le pratiquons dans WASABI.

Comme on vient de le voir, il est possible et nécessaire d'intégrer différents types de données dynamiques. Pourtant, le développement des algorithmes d'inférence de RRG est entré dans une logique de compétition très tôt, jusqu'à la création d'une compétition internationale (Challenge DREAM [102]) dont la pertinence est discutable (cf résultats sur la levure [100]). L'idée sous-jacente est que les données, ou plus précisément, le jeu de données transcriptomiques, contient toute l'information nécessaire pour reconstruire le réseau, charge aux développeurs d'implémenter l'algorithme le plus performant dans l'extraction de l'information cachée. A mon sens, il faut changer de paradigme. Les données expérimentales, quelque soit le type ou la quantité, ne contiennent intrinsèquement qu'une partie de l'information du système étudié, et aucun algorithme aussi sophistiqué soit-il ne pourra en retirer plus d'information. La question du biologiste qui cherche à identifier son système est donc : quelles données dois-je générer pour avoir le maximum d'information ? Cette question est loin d'être triviale, car les données ne doivent pas être redondantes et certaines ne révèlent leur information que lorsqu'elles sont croisées. Une réponse à cette question a été l'approche par perturbation, avec la génération de centaines de KO [25, 26]. Cependant, il faut aller plus loin dans l'intégration des données comme nous le proposons avec WASABI.

Le dernier point que je souhaiterai mettre en lumière dans cette partie est l'importance de la capitalisation des connaissances. Comme nous venons de le voir, il est important d'intégrer des données hétérogènes. Cependant, en pratique, les technologies expérimentales ne sont pas toujours disponibles ou abordables au même moment, et bien souvent les contraintes de coûts et délais imposent de les espacer temporellement. Il est donc important d'adopter une approche itérative basée sur des modèles évolutifs, dans le sens où l'on peut raffiner ces modèles en agrégeant des nouvelles données et hypothèses, en conservant les connaissances préalables.

Les modèles d'avion développés et améliorés pendant des décennies en aéronautique sont une démonstration de ce principe. En ce sens, les approches statistiques sont limitées car l'introduction de nouvelles variables nécessite la relance de l'analyse. De même, les approches par résolution mathématique imposent de modifier les équations de départ ce qui rend la solution existante obsolète. Par contre, c'est l'une des forces de l'approche WASABI qui peut facilement s'adapter pour intéuyigrer des nouvelles informations. Par exemple, dans sa version actuelle WASABI n'exploite pas les distributions jointes des matrices d'expression en cellule-unique, ce qui revient à délaisser une part importante de l'information. Comme nous le verrons dans les perspectives, il est parfaitement possible d'adapter l'algorithme à moindre frais. On retiendra que WASABI a été développé avec la volonté de s'inscrire dans la durée et servir de support ("framework") à l'intégration de la masse de données biologiques en plein développement exponentiel [111].

3.2 Nouvelle vision de l'organisation des RRG

3.2.1 La stochasticité, force motrice guidée par les RRG

Si nous reconnaissons la stochasticité biologique et l'intégrons dans notre démarche, il n'en n'est pas de même pour les autres approches comme nous l'avons vu en introduction. Le plus souvent, la variabilité, et surtout le nombre important de zéros dans les mesures en cellule unique, est attribué à des erreurs techniques communément appelées "drop out". Cependant, les résultats des travaux de l'article 1 ont confirmé d'une part que la stochasticité intrinsèque à l'expression génétique participait à la forte variabilité moléculaire inter-cellulaire, bien plus que le processus de différenciation et le bruit technique. D'autre part, cette variabilité évolue transitoirement au cours de la différenciation en atteignant un pic entre 8 et 24h, précédent l'engagement irréversible des cellules. Ces résultats suggèrent un rôle fonctionnel de l'évolution de la variabilité au cours du processus de différenciation qui a été validé plus tard par une autre équipe [112] et par nous-même [113] (nous reviendrons sur ce point dans l'étude des perspectives). C'est pourquoi il nous a semblé nécessaire de développer un modèle mécaniste stochastique de RRG dans l'article 2. Dans ce modèle on notera bien que ce sont les paramètres stochastiques k_{on}, k_{off} qui sont régulés. A notre connaissance, il s'agit du premier

modèle de RRG stochastique basé sur des hypothèses mécanistes utilisées pour l'inférence de RRG. .

Cependant, il ne faudrait pas penser que dans notre vision des RRG tout n'est qu'aléa. Dans notre vision la stochasticité est contrainte par les interactions entre gènes, mais elle est aussi filtrée par la stabilité (demi-vie) importante des protéines. Cela revient à compléter la vision déterministe des RRG avec la notion de probabilité. Ainsi, la topologie des interactions dans un RRG augmente la probabilité d'un certain état, mais elle autorise d'autres états à probabilités beaucoup plus faibles. L'évolution naturelle, qui ne peut se débarrasser du bruit moléculaire, aurait ainsi sélectionné des contraintes pour maîtriser et guider cette stochasticité pour définir des phénotypes stables. Mais elle pourrait aussi la mettre à profit lors d'un processus d'adaptation, comme la différenciation, en relâchant transitoirement la variabilité pour explorer l'espace des états génétiques possibles à la recherche d'un état en adéquation avec son nouvel environnement. Cette vision, qui s'apparente à un processus évolutif à l'échelle moléculaire sur une très courte période, a déjà été théorisée [114]. Des résultats récents appuient cette hypothèse, comme les expériences de Erez Braun [115] qui montrent que face à une situation très critique et complètement nouvelle où le RRG d'une fonction vitale est détourné, des levures sont capables en quelques heures de s'adapter et de trouver à chaque expérience une solution nouvelle en terme d'état génétique. La première interprétation de ces résultats est que le RRG est fortement dégénéré, c'est à dire qu'il existe de nombreuses interactions, certainement redondantes, qui autorisent une multi-stabilité très large. Dit autrement il existe de nombreux chemin pour aller à Rome [116], et même si le chemin le plus court et le plus large est coupé, la cellule peut emprunter des voies alternatives. De plus, comme dans nos travaux, les expériences de Braun font apparaître une augmentation transitoire de la variabilité, mais au niveau métabolique. D'autres travaux de notre équipe montrent aussi le lien entre différenciation, évolution de la stochasticité et métabolisme (manuscrit en préparation). Ensemble, ces observations suggèrent que la différenciation peut être vu comme un cas d'adaptation à un nouvel environnement qui génère un stress métabolique, qui a son tour induit un relâchement des contraintes sur le RRG pour autoriser la cellule à explorer son espace d'état génétique sous la contrainte des interactions existantes qui favorisent un certain état stable parmi d'autres. Une fois qu'un état adéquat est atteint, le stress

diminue et le RRG est de nouveau "verrouillé". Cette vision de la différenciation n'est qu'une spéculation, mais elle permet d'expliquer plusieurs observations et offre surtout l'avantage d'expliquer les propriétés incroyables d'adaptation des systèmes vivants qu'aucun modèle déterministe ne peut expliquer. Cette vision est importante car elle a des répercussions sur la topologie des RRG que nous allons aborder.

3.2.2 Une topologie de RRG originale

Je reviens ici sur les caractéristiques typiques des RRG candidats générés par WASABI dans l'article 3 à partir des données *in vitro* générées dans l'article 1 sur notre modèle biologique de différenciation érythrocytaire aviaire.

Le stimulus joue un rôle central dans les RRG candidats. Il affecte directement un nombre important de gènes dit "early" en les inhibant en majorité. En soit il s'agit d'un résultat important car très souvent dans la littérature le stimulus n'est pas représenté. Néanmoins, il se peut que cette influence importante du stimulus sur notre RRG soit due à la nature même du milieu qui le compose qui est issu de sérum de poulets anémiés. Ce sérum peut contenir un nombre important de molécules de signalisation comme des cytokines. De plus, cette position centrale du stimulus impose une topologie très parallèle du RRG, ce qui, combiné à l'absence de gènes centrales communément désignés par "hub", est à l'opposé de la vision classique répandue de RRG organisés en "small world" [117]. Cependant il faut noter que dans la version actuelle de WASABI les interactions redondantes ne sont pas considérées, c'est à dire qu'un gène ne peut être régulé que par un gène, sauf en cas de rétro-contrôle. Cette limitation peut donc expliquer la structure très parallèle du RRG, nous discuterons des perspectives d'évolution de WASABI sur ce point dans la partie suivante. La profondeur des RRG, c'est à dire le nombre maximum d'interactions entre le stimulus et un gène, est limité à 5 interactions. Cette limitation est logique et en accord avec l'inertie moyenne des gènes et le temps de différenciation. Les protéines ont une demi-vie d'environ 15h en considérant la dilution due aux divisions cellulaires, et le processus de différenciation est supposé achevé à 72h, ce qui laisse le temps en première approximation à 4 ou 5 interactions comme on l'observe. Ce résultat peut difficilement être remis en cause si on s'accorde sur les estimations des demi-vies des ARN et des protéines. Or, dans la littérature, la plupart des RRG arborent

des profondeurs de réseau très importantes à cause du fort degré de connectivité, ce qui nous laisse très sceptique vis à vis de leur pertinence biologique. Cette limitation de la profondeur des RRG nous a amené à poser la question du transport de l'information dans ces RRG très bruités. Des travaux non publiés dans l'équipe ont montré que l'information ne pouvait se propager à plus de 4 ou 5 gènes en l'absence d'auto-activation, ce qui va dans le sens de la pertinence biologique de RRG parallèle à profondeur réduite. Cependant, ce constat soulève la problématique du maintien et de la transmission de l'information lors d'étape successives de différenciation comme une cellule souche en connait. Cette contradiction peut être levée par la présence de boucles positives comme nous allons le détailler.

Les boucles positives sont connues pour avoir un rôle important dans les réseaux car elles permettent l'amplification, la création de mémoire et la multi-stabilité. C'est pourquoi nous avons développé dans l'article 2 un estimateur de coefficient d'auto-activation basé sur l'analyse du transitoire des distributions marginales. En quelques mots, il s'agit de détecter de la bi-stabilité dans les distributions engendrées par une potentielle boucle d'activation positive, qui n'est pas nécessairement de l'auto-activation directe. Cet estimateur est utilisé par WASABI dans l'article 3 sur les données *in vitro*. De façon très surprenante, plus de 80% des gènes seraient sous l'influence d'une boucle positive. Si la présence de boucles positives étaient attendues, une telle proportion est plus étonnante. Habituellement les auto-activations sont peu représentées dans la littérature, cependant ce résultat a été confirmé par une étude récente sur des cellules souches embryonnaires [118]. Une première interprétation, en lien avec le paragraphe précédent, est la capacité de ces RRG avec de nombreuses boucles positives à transporter l'information malgré la stochasticité intrinsèque. Une fois qu'un gène est activé, la boucle positive l'active à son niveau maximum ce qui lui permet de relayer efficacement l'information. De plus, il mémorise l'information et devient autonome, ce qui permet d'expliquer facilement l'irréversibilité de l'engagement des cellules différenciées observée dans l'article 1. Ce résultat peut expliquer en partie pourquoi les modèles booléens, malgré leur extrême simplicité, peuvent être biologiquement pertinent si tous les gènes sont fortement auto-activés. Par contre, les approches par corrélation et indépendance statistique auront plus de difficultés à inférer les régulateurs d'un gène auto-activé car celui-ci devient autonome vis à vis de ces régulateurs. La présence importante des boucles positives peut

aussi s'interpréter à la lumière de la vision stochastique et dégénérée des RRG que je viens d'exposer précédemment. La création d'une multitude d'états stables sont autant de solutions possibles qui facilitent la mémorisation de la solution.

3.3 Perspectives

3.3.1 Amélioration de la pertinence biologique du modèle mécaniste de RRG

Lors de la définition du modèle mécaniste de RRG nous avons écarté plusieurs hypothèses biologiques connues sur la régulation de l'expression génétique et nous avons rajouté des contraintes. La contrainte la plus importante est la limitation a un seul régulateur possible par itération dans WASABI. Cette limitation est sévère et doit être levée pour deux raisons. La première est que l'information contenue dans les distributions jointes n'est pas exploitée par WASABI. On pourrait adapter un algorithme d'inférence comme celui développé dans l'article 2 pour proposer à chaque étape itérative de l'inférence des interactions candidates coopératives ou redondantes en accord avec les distributions jointes. On peut réutiliser la métrique de Kantorovitch pour calculer la distance entre 2 distributions jointes, mais en pratique le temps de calcul est polynomial par rapport au nombre de gène [119] ce qui rend l'exercice impossible. Cependant, nous avons réussi à partir de notre modèle de RRG à dériver une approximation de la distribution jointe avec beaucoup moins de paramètres [120], ce qui rend le calcul faisable. La deuxième raison est la pertinence biologique des interactions redondantes qui créé de la dégénérescence. Comme je l'ai développé dans la nouvelle vision des RRG que nous proposons, les propriétés d'adaptabilités des RRG sont très certainement liées au degré de dégénérescence. Dans l'optique de l'utilisation des RRG inférés pour réaliser des prédictions, il est donc important de reproduire ces propriétés.

Un autre axe d'amélioration est l'ajout de nouveaux types d'interactions, comme la régulation du taux de dégradation des ARN par des ARN non codant, ou la régulation du taux de dégradation des protéines par d'autres protéines. Nous avons vu que les taux de dégradations estimés à partir de données expérimentales varient au cours du temps. Il y a donc très certainement une régulation et il serait intéressant d'inférer les causalités. Le modèle actuel n'explique pas ces régulations, mais il reproduit leur effet. Le modèle est en ce sens phéno-

ménologique, mais pas explicatif. La modification du modèle serait simple à réaliser et elle ne remettrait pas en cause la stratégie d'inférence de WASABI, ce qui implique que l'impact sur le code serait mineur. Cependant, l'inférence de ces nouvelles régulations exigerait très certainement la production de nouvelles données expérimentales.

A plus long terme, il faudrait également poser la question de la modélisation de l'environnement chromatinien. Comme nous l'avons exposé dans la vision des RRG, nos résultats suggèrent un relâchement global des contraintes au niveau du RRG lors de la différenciation, suivi d'un verrouillage pour stabiliser le nouvel état. Des observations de la variation de l'état chromatinien au cours de processus de différenciation [121, 122] montrent que la chromatine est plus ouverte et permissive à l'état souche et au cours de la différenciation. De plus, il a été établi un lien fonctionnel entre d'une part l'augmentation de la variabilité de l'expression génétique par déstabilisation de la chromatine à l'aide de drogues, et d'autre part l'augmentation de la vitesse du processus de différenciation dans des cellules ES de souris [112] ainsi que sur notre modèle biologique des cellules T2EC [113].

3.3.2 Amélioration des performances de l'inférence itérative

Les performances de l'inférence sont liées à la qualité et la quantité des données expérimentales mais aussi au rapport entre la puissance de calcul et la gestion de la combinatoire. En effet, la qualité des RRG candidats retournés par WASABI est directement liée à ce rapport. Si le nombre d'interactions à tester pour l'ensemble des RRG candidats lors d'une itération est supérieur au nombre de machines disponibles, il faut fatalement exclure des interactions. C'est ce que nous faisons actuellement en sélectionnant les premiers RRG candidats selon leur qualité de recoupement. Le problème est que les premières itérations de WASABI sont peu discriminantes et il y a un risque d'éliminer arbitrairement de bons candidats. C'est pourquoi il est nécessaire de limiter les interactions possibles et d'augmenter la capacité de calcul parallèle.

Pour limiter d'avantage le nombre d'interactions possibles il y a 2 axes d'amélioration. Le premier est l'augmentation du nombre de points expérimentaux dans les cinétiques pour raffiner l'estimation du temps de vagues des promoteurs et protéines. La fenêtre temporelle que l'on considère aujourd'hui est très permissive, entre -20h et +30h, car elle a été calibrée

en considérant un nombre restreint de point expérimentaux. Il serait nécessaire de doubler le nombre de point pour passer à une douzaine. L'autre axe est la prise en compte des distributions jointes comme nous l'avons abordé ci-dessus. Cela permettrait de limiter les interactions possibles en combinant les contraintes temporelles de synchronisation existantes aux contraintes imposées par les distributions jointes. De plus, le recoupement des données simulées avec les données expérimentales ne se ferait plus sur la base des distributions marginales mais jointes, ce qui contraindrait plus le recoupement. Toutefois, il faudrait un nombre de cellules beaucoup plus important pour travailler sur les distributions jointes, ce qui impose de passer à des technologies de plus haut débit pour le nombre de cellule. Ces technologies existent déjà avec l'utilisation de gouttelettes [123] ou de "bar coding" des cellules avec la méthode du SPLIT-seq [124].

Pour accroître la capacité de calcul HPC on peut tout simplement augmenter le nombre de machines et leur puissance de calcul, mais on peut aussi optimiser leur utilisation comme nous l'avons abordé dans l'article 4. Nous utilisons actuellement 400 machines mis à notre disposition gracieusement par le centre de calcul de l'IN2P3. Nous sommes déjà limités par ce nombre et on estime qu'il faudrait quelques milliers de machine pour éviter de rejeter des bons candidats. Cette solution demande un investissement financier non négligeable, et on sait d'avance que si l'on conserve la gestion actuelle du flux des tâches plus de la moitié des machines ne seront pas exploitées sur l'ensemble de l'inférence. Ce problème vient de la variabilité des temps de calibration en fonction de la topologie des RRG candidats. En particulier, les RRG avec beaucoup d'interactions croisées et une plus grande profondeur de réseau seront plus long à calibrer. Une solution serait donc de ne pas attendre la fin de toutes les tâches de calibration lors d'une itération, mais de poursuivre les itérations sur les RRG candidats les plus rapides en termes de calibration. Cependant cette solution amène d'autres problématiques qu'il faudrait étudier en détails avec une équipe spécialisée dans la gestion HPC.

3.3.3 Applications potentielles de WASABI

L'objectif de l'inférence des RRG est d'utiliser les réseaux candidats pour orienter la recherche, notamment pour l'étude de pathologies. Certaines équipes pratiquent déjà ce type

d'approche comme l'équipe d'Andrea Califano qui exploite l'algorithme ARACNE [26, 45]. WASABI a été développé dans la perspective de l'appliquer sur des problématiques biologiques faisant intervenir des interactions génétiques. L'objectif à long terme serait de maîtriser suffisamment les RRG pour pouvoir guider les cellules vers un état désiré. En médecine régénérative l'objectif serait de reprogrammer les cellules vers un nouveau phénotype, comme dans ces travaux [125, 126] où des cellules de l'épiderme sont reprogrammées en cellules souches hématopoïétique. Le cancer peut aussi être une application potentielle avec la volonté de guider les cellules tumorales vers un état sain, neutre ou la mort cellulaire. Les thérapies géniques sont aussi une application évidente tout comme la biologie synthétique qui implique aussi une reprogrammation des RRG. La contribution de WASABI dans ces problématiques serait l'identification de nouvelle cibles thérapeutiques, la prédiction de l'efficacité de combinaison de drogues ainsi que leur robustesse et effets secondaires. On peut aussi penser à la définition de nouveaux marqueurs pour du diagnostic.

Pour réaliser ces prédictions WASABI devra être amélioré comme je viens de le décrire ci-dessus, mais malgré toutes les améliorations possibles il ne pourra certainement pas donner une réponse précise à partir d'une seule série d'expérience. Cette imprécision fondamentale vient de la nature stochastique des RRG et de leur forte dégénérescence (non-identifiabilité). C'est pourquoi WASABI propose une liste de RRG candidats qui peuvent être potentiellement très différents. Il est donc nécessaire, comme je l'ai détaillé plus haut, de capitaliser les informations à partir de plusieurs expériences plutôt que d'améliorer asymptotiquement les algorithmes. WASABI doit donc s'inscrire dans une démarche itérative au sens général de la biologie des systèmes, c'est à dire la définition rationnelle de nouvelles expériences les plus informatives à partir de l'analyse d'expériences précédente. Cette démarche, dite de "Design Of Experiment", existe déjà et de nombreux travaux ont été réalisé sur la base de RRG déterministe [127, 128]. Dans le cas de WASABI il faudra adapter ces approches pour proposer de nouvelles expériences qui permettront de discriminer au maximum entre les RRG candidats pour diminuer la liste et perfectionner les candidats sélectionnés.

J'espère avec ces travaux avoir apporté une pierre à l'édifice d'une solution à la problématique d'inférence des RRG. L'objectif visé est de proposer aux biologistes des outils intégratifs et évolutifs basés sur des concepts d'ingénierie. Même si WASABI doit être encore

amélioré, il s'inscrit dans une perspective à long terme qui parie sur l'évolution exponentielle des technologies en biologie cellulaire et moléculaire. Seule une intégration croisée des données et connaissances, actuelles et futures, permettra de faire avancer pas à pas, mais significativement, notre compréhension et maîtrise des systèmes biologiques complexes.

Bibliographie

[1] Prokopiou, S., Barbarroux, L., Bernard, S., Mafille, J., Leverrier, Y., Arpin, C., Marvel, J., Gandrillon, O., and Crauste, F. Multiscale modeling of the early cd8 t cell immune response in lymph nodes : an integrative study. *Computation*, 2 :159–181, 2014.

[2] Mercer, T. R., Dinger, M. E., and Mattick, J. S. Long non-coding rnas : insights into functions. *Nature Reviews Genetics*, 10(3) :155, 2009.

[3] He, L. and Hannon, G. J. Micrornas : small rnas with a big role in gene regulation. *Nature Reviews Genetics*, 5(7) :522, 2004.

[4] Davidson, E. H. *The regulatory genome : gene regulatory networks in development and evolution*. Elsevier, 2010.

[5] Valencia-Sanchez, M. A., Liu, J., Hannon, G. J., and Parker, R. Control of translation and mrna degradation by mirnas and sirnas. *Genes & development*, 20(5) :515–524, 2006.

[6] Olsen, J. V. and Mann, M. Status of large-scale analysis of post-translational modifications by mass spectrometry. *Mol Cell Proteomics*, 12(12) :3444–52, 2013.

[7] Manning, K. S. and Cooper, T. A. The roles of rna processing in translating genotype to phenotype. *Nat Rev Mol Cell Biol*, 18(2) :102–114, 2017.

[8] Panning, B. and Jaenisch, R. Rna and the epigenetic regulation of x chromosome inactivation. *Cell*, 93(3) :305–308, 1998.

[9] Rousseau, G. G. Interaction of steroids with hepatoma cells : molecular mechanisms of glucocorticoid hormone action. *Journal of steroid biochemistry*, 6(1) :75–89, 1975.

[10] Benenson, Y. Biomolecular computing systems : principles, progress and potential. *Nature Reviews Genetics*, 13 :455, 2012.

[11] Tyson, J. J., Hong, C. I., Thron, C. D., and Novak, B. A simple model of circadian rhythms based on dimerization and proteolysis of per and tim. *Biophysical journal*, 77(5) :2411–2417, 1999.

[12] Mattick, J. S., Taft, R. J., and Faulkner, G. J. A global view of genomic information–moving beyond the gene and the master regulator. *Trends in genetics*, 26(1) :21–28, 2010.

[13] Gandrillon, O., Schmidt, U., Beug, H., and Samarut, J. Tgf-β cooperates with tgf-α to induce the self–renewal of normal erythrocytic progenitors : evidence for an autocrine mechanism. *The EMBO Journal*, 18(10) :2764–2781, 1999.

[14] Orkin, S. H. and Zon, L. I. Hematopoiesis : an evolving paradigm for stem cell biology. *Cell*, 132(4) :631–644, 2008.

[15] Jacob, F. and Monod, J. On regulation of gene activity. *Cold Spring Harbor Symposia on Quantitative Biology*, 26 :193, 1961.

[16] Levine, M. and Davidson, E. H. Gene regulatory networks for development. *Proc Natl Acad Sci U S A*, 102(14) :4936–42, 2005.

[17] Takahashi, K. and Yamanaka, S. Induction of pluripotent stem cells from mouse embryonic and adult fibroblast cultures by defined factors. *cell*, 126(4) :663–676, 2006.

[18] Yamanaka, S. Induced pluripotent stem cells : Past, present, and future. *Cell Stem Cell*, 10(6) :678 – 684, 2012.

[19] Jopling, C., Boue, S., and Belmonte, J. C. I. Dedifferentiation, transdifferentiation and reprogramming : three routes to regeneration. *Nature reviews Molecular cell biology*, 12(2) :79, 2011.

[20] Mojtahedi, M., Skupin, A., Zhou, J., Castaño, I. G., Leong-Quong, R. Y. Y., Chang, H., Trachana, K., Giuliani, A., and Huang, S. Cell fate decision as high-dimensional critical state transition. *PLOS Biology*, 14(12) :1–28, 12 2016.

[21] Kupiec, J. J. A darwinian theory for the origin of cellular differentiation. *Molecular and General Genetics MGG*, 255(2) :201–208, Jun 1997.

[22] Schena, M., Shalon, D., Davis, R. W., and Brown, P. O. Quantitative monitoring of gene expression patterns with a complementary dna microarray. *Science*, 270(5235) :467–470, 1995.

[23] Lockhart, D. J., Dong, H., Byrne, M. C., Follettie, M. T., Gallo, M. V., Chee, M. S., Mittmann, M., Wang, C., Kobayashi, M., and Norton, H. Expression monitoring by

hybridization to high-density oligonucleotide arrays. *Nature biotechnology*, 14(13) :1675, 1996.

[24] Gardner, T. S., Di Bernardo, D., Lorenz, D., and Collins, J. J. Inferring genetic networks and identifying compound mode of action via expression profiling. *Science*, 301(5629) :102–105, 2003.

[25] Hughes, T. R., Marton, M. J., Jones, A. R., Roberts, C. J., Stoughton, R., Armour, C. D., Bennett, H. A., Coffey, E., Dai, H., and He, Y. D. Functional discovery via a compendium of expression profiles. *Cell*, 102(1) :109–126, 2000.

[26] Basso, K., Margolin, A. A., Stolovitzky, G., Klein, U., Dalla-Favera, R., and Califano, A. Reverse engineering of regulatory networks in human b cells. *Nature genetics*, 37(4) :382, 2005.

[27] Spellman, P. T., Sherlock, G., Zhang, M. Q., Iyer, V. R., Anders, K., Eisen, M. B., Brown, P. O., Botstein, D., and Futcher, B. Comprehensive identification of cell cycle–regulated genes of the yeast saccharomyces cerevisiae by microarray hybridization. *Molecular biology of the cell*, 9(12) :3273–3297, 1998.

[28] Arbeitman, M. N., Furlong, E. E., Imam, F., Johnson, E., Null, B. H., Baker, B. S., Krasnow, M. A., Scott, M. P., Davis, R. W., and White, K. P. Gene expression during the life cycle of drosophila melanogaster. *Science*, 297(5590) :2270–2275, 2002.

[29] Bittner, M., Meltzer, P., Chen, Y., Jiang, Y., Seftor, E., Hendrix, M., Radmacher, M., Simon, R., Yakhini, Z., and Ben-Dor, A. Molecular classification of cutaneous malignant melanoma by gene expression profiling. *Nature*, 406(6795) :536, 2000.

[30] Morin, R. D., Bainbridge, M., Fejes, A., Hirst, M., Krzywinski, M., Pugh, T. J., McDonald, H., Varhol, R., Jones, S. J., and Marra, M. A. Profiling the hela s3 transcriptome using randomly primed cdna and massively parallel short-read sequencing. *Biotechniques*, 45(1) :81, 2008.

[31] Friedman, N., Linial, M., Nachman, I., and Pe'er, D. Using bayesian networks to analyze expression data. *Journal of computational biology*, 7(3-4) :601–620, 2000.

[32] Pe'er, D., Regev, A., Elidan, G., and Friedman, N. Inferring subnetworks from perturbed expression profiles. *Bioinformatics*, 17(suppl 1) :S215–S224, 2001.

[33] Dojer, N., Gambin, A., Mizera, A., Wilczyński, B., and Tiuryn, J. Applying dynamic bayesian networks to perturbed gene expression data. *BMC bioinformatics*, 7(1) :249, 2006.

[34] Chai, L. E., Mohamad, M. S., Deris, S., Chong, C. K., Choon, Y. W., Ibrahim, Z., and Omatu, S. Inferring gene regulatory networks from gene expression data by a dynamic bayesian network-based model. In *Distributed Computing and Artificial Intelligence*, pages 379–386. Springer, 2012.

[35] Yu, J., Smith, V. A., Wang, P. P., Hartemink, A. J., and Jarvis, E. D. Advances to bayesian network inference for generating causal networks from observational biological data. *Bioinformatics*, 20(18) :3594–603, 2004.

[36] Kunga, T. A. and Mohamada, M. S. Using bayesian networks to construct gene regulatory networks from microarray data. *Jurnal Teknologi*, 1, 2012.

[37] Wu, H. and Liu, X. Dynamic bayesian networks modeling for inferring genetic regulatory networks by search strategy : Comparison between greedy hill climbing and mcmc methods. In *Proc. of World Academy of Science, Engineering and Technology*, volume 34, pages 224–234, 2008.

[38] Werhli, A. V. and Husmeier, D. Reconstructing gene regulatory networks with bayesian networks by combining expression data with multiple sources of prior knowledge. *Statistical applications in genetics and molecular biology*, 6(1), 2007.

[39] Yavari, F., Towhidkhah, F., and Gharibzadeh, S. Gene regulatory network modeling using bayesian networks and cross correlation. In *2008 Cairo International Biomedical Engineering Conference*, pages 1–4.

[40] Vinh, N. X., Chetty, M., Coppel, R., and Wangikar, P. P. Gene regulatory network modeling via global optimization of high-order dynamic bayesian network. *BMC bioinformatics*, 13(1) :131, 2012.

[41] Yang, B., Zhang, J., Shang, J., and Li, A. A bayesian network based algorithm for gene regulatory network reconstruction. In *2011 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC)*, pages 1–4, Sept 2011.

[42] Grzegorczyk, M. and Husmeier, D. Improvements in the reconstruction of time-varying gene regulatory networks : dynamic programming and regularization by information sharing among genes. *Bioinformatics*, 27(5) :693–699, 2010.

[43] Butte, A. J. and Kohane, I. S. *Mutual information relevance networks : functional genomic clustering using pairwise entropy measurements*, pages 418–429. World Scientific, 1999.

[44] Steuer, R., Kurths, J., Daub, C. O., Weise, J., and Selbig, J. The mutual information : detecting and evaluating dependencies between variables. *Bioinformatics*, 18(suppl 2) :S231–S240, 2002.

[45] Margolin, A. A., Nemenman, I., Basso, K., Wiggins, C., Stolovitzky, G., Dalla Favera, R., and Califano, A. Aracne : an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC bioinformatics*, 7 :S7, 2006.

[46] Kauffman, S. A. Metabolic stability and epigenesis in randomly constructed genetic nets. *Journal of theoretical biology*, 22(3) :437–467, 1969.

[47] Silvescu, A. and Honavar, V. Temporal boolean network models of genetic networks and their inference from gene expression time series. *Complex Systems*, 13(1) :61–78, 2001.

[48] Hickman, G. J. and Hodgman, T. C. Inference of gene regulatory networks using boolean-network inference methods. *Journal of bioinformatics and computational biology*, 7(06) :1013–1029, 2009.

[49] Albert, R. *Boolean modelingof genetic regulatory networks*, pages 459–481. Springer, 2004.

[50] Harvey, I. and Bossomaier, T. Time out of joint : Attractors in asynchronous random boolean networks. In *Proceedings of the Fourth European Conference on Artificial Life*, pages 67–75. MIT Press, Cambridge.

[51] Apostel, J. L. Classification of random boolean networks. *Artificial Life 8*, 8 :1, 2003.

[52] Shmulevich, I., Dougherty, E. R., Kim, S., and Zhang, W. Probabilistic boolean networks : a rule-based uncertainty model for gene regulatory networks. *Bioinformatics*, 18(2) :261–274, 2002.

[53] Ching, W.-K., Zhang, S.-Q., Jiao, Y., Akutsu, T., and Wong, A. Optimal finite-horizon control for probabilistic boolean networks with hard constraints. *Lecture Notes in Operations Research 7*, 2007.

[54] Marshall, S., Yu, L., Xiao, Y., and Dougherty, E. R. Inference of a probabilistic boolean network from a single observed temporal sequence. *EURASIP Journal on Bioinformatics and Systems Biology*, 2007 :5–5, 2007.

[55] Liang, S., Fuhrman, S., and Somogyi, R. Reveal, a general reverse engineering algorithm for inference of genetic network architectures. 1998.

[56] Akutsu, T., Miyano, S., and Kuhara, S. *Identification of genetic networks from a small number of gene expression patterns under the Boolean network model*, pages 17–28. World Scientific, 1999.

[57] Akutsu, T., Miyano, S., and Kuhara, S. Inferring qualitative relations in genetic networks and metabolic pathways. *Bioinformatics*, 16(8) :727–734, 2000.

[58] Hakamada, K., Hanai, T., Honda, H., and Kobayashi, T. A preprocessing method for inferring genetic interaction from gene expression data using boolean algorithm. *Journal of bioscience and bioengineering*, 98(6) :457–463, 2004.

[59] Zhao, W., Serpedin, E., and Dougherty, E. R. Inferring gene regulatory networks from time series data using the minimum description length principle. *Bioinformatics*, 22(17) :2129–2135, 2006.

[60] Ching, W.-K., Zhang, S., Ng, M. K., and Akutsu, T. An approximation method for solving the steady-state probability distribution of probabilistic boolean networks. *Bioinformatics*, 23(12) :1511–1518, 2007.

[61] Wimburly, F. C., Heiman, T., Ramsey, J., and Glymour, C. Experiments on the accuracy of algorithms for inferring the structure of genetic regulatory networks from microarray expression levels. 2003.

[62] di Bernardo, D., Thompson, M. J., Gardner, T. S., Chobot, S. E., Eastwood, E. L., Wojtovich, A. P., Elliott, S. J., Schaus, S. E., and Collins, J. J. Chemogenomic profiling on a genome-wide scale using reverse-engineered gene networks. *Nature biotechnology*, 23(3) :377, 2005.

[63] Bansal, M., Gatta, G. D., and Di Bernardo, D. Inference of gene regulatory networks and compound mode of action from time course gene expression profiles. *Bioinformatics*, 22(7) :815–822, 2006.

[64] Ando, S., Sakamoto, E., and Iba, H. Evolutionary modeling and inference of gene network. *Information Sciences*, 145(3-4) :237–259, 2002.

[65] Chen, K.-C., Wang, T.-Y., Tseng, H.-H., Huang, C.-Y. F., and Kao, C.-Y. A stochastic differential equation model for quantifying transcriptional regulatory network in saccharomyces cerevisiae. *Bioinformatics*, 21(12) :2883–2890, 2005.

[66] Polynikis, A., Hogan, S., and di Bernardo, M. Comparing different ode modelling approaches for gene regulatory networks. *Journal of theoretical biology*, 261(4) :511–530, 2009.

[67] Warren, L., Bryder, D., Weissman, I. L., and Quake, S. R. Transcription factor profiling in individual hematopoietic progenitors by digital rt-pcr. *Proc Natl Acad Sci U S A*, 103(47) :17807–12, 2006.

[68] Kolodziejczyk, A. A., Kim, J. K., Svensson, V., Marioni, J. C., and Teichmann, S. A. The technology and biology of single-cell rna sequencing. *Mol Cell*, 58(4) :610–20, 2015.

[69] Clark, S. J., Lee, H. J., Smallwood, S. A., Kelsey, G., and Reik, W. Single-cell epigenomics : powerful new methods for understanding gene regulation and cell identity. *Genome Biology*, 17, 2016.

[70] Levsky, J. M. and Singer, R. H. Gene expression and the myth of the average cell. *Trends in cell biology*, 13(1) :4–6, 2003.

[71] Pierson, E. and Yau, C. Zifa : Dimensionality reduction for zero-inflated single-cell gene expression analysis. *Genome biology*, 16(1) :241, 2015.

[72] Fiers, M., Minnoye, L., Aibar, S., Bravo Gonzalez-Blas, C., Kalender Atak, Z., and Aerts, S. Mapping gene regulatory networks from single-cell omics data. *Brief Funct Genomics*, 2018.

[73] Babtie, A., Chan, T. E., and Stumpf, M. P. H. Learning regulatory models for cell development from single cell transcriptomic data. *Current Opinion in Systems Biology*, 5 :72D81, 2017.

[74] Chen, S. and Mar, J. C. Evaluating methods of inferring gene regulatory networks highlights their lack of performance for single cell gene expression data. *BMC Bioinformatics*, 19(1) :232, 2018.

[75] Brennecke, P., Anders, S., Kim, J. K., Kołodziejczyk, A. A., Zhang, X., Proserpio, V., Baying, B., Benes, V., Teichmann, S. A., Marioni, J. C., et al. Accounting for technical noise in single-cell rna-seq experiments. *Nature methods*, 10(11) :1093, 2013.

[76] Cannoodt, R., Saelens, W., and Saeys, Y. Computational methods for trajectory inference from single-cell transcriptomics. *Eur J Immunol*, 46(11) :2496–2506, 2016.

[77] Chen, H., Guo, J., Mishra, S. K., Robson, P., Niranjan, M., and Zheng, J. Single-cell transcriptional analysis to uncover regulatory circuits driving cell fate decisions in early mouse development. *Bioinformatics*, 31(7) :1060–6, 2015.

[78] Lim, C. Y., Wang, H., Woodhouse, S., Piterman, N., Wernisch, L., Fisher, J., and Gottgens, B. Btr : training asynchronous boolean models using single-cell expression data. *BMC Bioinformatics*, 17(1) :355, 2016.

[79] Moignard, V., Woodhouse, S., Haghverdi, L., Lilly, A. J., Tanaka, Y., Wilkinson, A. C., Buettner, F., Macaulay, I. C., Jawaid, W., Diamanti, E., Nishikawa, S., Piterman, N., Kouskoff, V., Theis, F. J., Fisher, J., and Gottgens, B. Decoding the regulatory network of early blood development from single-cell gene expression measurements. *Nat Biotechnol*, 33(3) :269–76.

[80] Matsumoto, H. and Kiryu, H. Scoup : a probabilistic model based on the ornstein-uhlenbeck process to analyze single-cell expression data during differentiation. *BMC Bioinformatics*, 17(1) :232, 2016.

[81] Cordero, P. and Stuart, J. M. *Tracing co-regulatory network dynamics in noisy, single-cell transcriptome trajectories*, pages 576–587. World scientific, 2016.

[82] Sanchez-Castillo, M., Blanco, D., Tienda-Luna, I. M., Carrion, M. C., and Huang, Y. A bayesian framework for the inference of gene regulatory networks from time and pseudo-time series data. *Bioinformatics*, 2017.

[83] Elowitz, M. B., Levine, A. J., Siggia, E. D., and Swain, P. S. Stochastic gene expression in a single cell. *Science*, 297(5584) :1183–1186, 2002. p39.

[84] Corre, G., Stockholm, D., Arnaud, O., Kaneko, G., Viñuelas, G., Yamagata, Y., Neildez-Nguyen, T. M. A., Kupiec, J.-J., Beslon, G., Gandrillon, O., and Paldi, A. Stochastic fluctuations and distributed control of gene expression impact cellular memory. *Plos ONE*, 0115574, 2014.

[85] Suter, D. M., Molina, N., Gatfield, D., Schneider, K., Schibler, U., and Naef, F. Mammalian genes are transcribed with widely different bursting kinetics. *Science*, 332(6028) :472–4, 2011. p39.

[86] Kim, J. K. and Marioni, J. C. Inferring the kinetics of stochastic gene expression from single-cell rna-sequencing data. *Genome Biol*, 14(1) :R7, 2013.

[87] Larson, D. R., Singer, R. H., and Zenklusen, D. A single molecule view of gene expression. *Trends Cell Biol*, 19(11) :630–7, 2009.

[88] Golding, I., Paulsson, J., Zawilski, S. M., and Cox, E. C. Real-time kinetics of gene activity in individual bacteria. *Cell*, 123(6) :1025–36, 2005. p39.

[89] Pedraza, J. M. and Paulsson, J. Effects of molecular memory and bursting on fluctuations in gene expression. *Science*, 319(5861) :339–43, 2008. p39.

[90] Papili Gao, N., Ud-Dean, S. M. M., Gandrillon, O., and Gunawan, R. Sincerities : Inferring gene regulatory networks from time-stamped single cell transcriptional expression profiles. *Bioinformatics*, 2017.

[91] Blasi, M. F., Casorelli, I., Colosimo, A., Blasi, F. S., Bignami, M., and Giuliani, A. A recursive network approach can identify constitutive regulatory circuits in gene expression data. *Physica A : Statistical Mechanics and its Applications*, 348 :349 – 370, 2005.

[92] Lee, W.-P. and Yang, K.-C. A clustering-based approach for inferring recurrent neural networks as gene regulatory networks. *Neurocomputing*, 71(4) :600 – 610, 2008. Neural Networks : Algorithms and Applications 50 Years of Artificial Intelligence : a Neuronal Approach.

[93] Xu, R., Venayagamoorthy, G. K., and Wunsch, D. C. Modeling of gene regulatory networks with hybrid differential evolution and particle swarm optimization. *Neural Networks*, 20(8) :917 – 927, 2007.

[94] Stoeckius, M., Hafemeister, C., Stephenson, W., Houck-Loomis, B., Chattopadhyay, P. K., Swerdlow, H., Satija, R., and Smibert, P. Simultaneous epitope and transcriptome measurement in single cells. *Nat Methods*, 14(9) :865–868, 2017.

[95] Wani, N. and Raza, K. Integrative approaches to reconstruct regulatory networks from multi-omics data : A review of state-of-the-art methods. 2018.

[96] Abduallah, Y. and Wang, J. T. L. A time-delayed information-theoretic approach to the reverse engineering of gene regulatory networks using apache spark. In *2017 IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing, 15th Intl Conf on Pervasive Intelligence and Computing, 3rd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress(DASC/PiCom/DataCom/CyberSciTech)*, pages 1106–1113.

[97] Noman, N., Palafox, L., and Iba, H. Reconstruction of gene regulatory networks from gene expression data using decoupled recurrent neural network model. In Suzuki, Y. and Nakagaki, T., editors, *Natural Computing and Beyond*, pages 93–103, Tokyo, 2013. Springer Japan.

[98] Chai, L. E., Loh, S. K., Low, S. T., Mohamad, M. S., Deris, S., and Zakaria, Z. A review on the computational approaches for gene regulatory network construction. *Comput Biol Med*, 48 :55–65, 2014. p40.

[99] Hecker, M., Lambeck, S., Toepfer, S., Van Someren, E., and Guthke, R. Gene regulatory network inference : data integration in dynamic models—a review. *Biosystems*, 96(1) :86–103, 2009.

[100] Marbach, D., Costello, J. C., Kuffner, R., Vega, N. M., Prill, R. J., Camacho, D. M., Allison, K. R., Consortium, D., Kellis, M., Collins, J. J., and Stolovitzky, G. Wisdom of crowds for robust gene network inference. *Nat Methods*, 9(8) :796–804, 2012.

[101] Moignard, V., Woodhouse, S., Haghverdi, L., Lilly, A. J., Tanaka, Y., Wilkinson, A. C., Buettner, F., Macaulay, I. C., Jawaid, W., Diamanti, E., Nishikawa, S., Piterman, N., Kouskoff, V., Theis, F. J., Fisher, J., and Gottgens, B. Decoding the regulatory network of early blood development from single-cell gene expression measurements. *Nat Biotechnol*, 33(3) :269–76, 2015. p40.

[102] Stolovitzky, G., Monroe, D., and Califano, A. Dialogue on reverse-engineering assessment and methods. *Annals of the New York Academy of Sciences*, 1115(1) :1–22, 2007.

[103] Matsumoto, H., Kiryu, H., Furusawa, C., Ko, M. S. H., Ko, S. B. H., Gouda, N., Hayashi, T., and Nikaido, I. Scode : an efficient regulatory network inference algorithm from single-cell rna-seq during differentiation. *Bioinformatics*, 33(15) :2314–2321, 2017.

[104] Ocone, A., Haghverdi, L., Mueller, N. S., and Theis, F. J. Reconstructing gene regulatory dynamics from high-dimensional single-cell snapshot data. *Bioinformatics*, 31(12) :i89–i96, 2015.

[105] Gallego, M. and Virshup, D. M. Post-translational modifications regulate the ticking of the circadian clock. *Nat Rev Mol Cell Biol*, 8(2) :139–48, 2007.

[106] Kærn, M., Elston, T. C., Blake, W. J., and Collins, J. J. Stochasticity in gene expression : from theories to phenotypes, Jun 1997.

[107] Huang, S. Systems biology of stem cells : three useful perspectives to help overcome the paradigm of linear pathways. *Philosophical Transactions of the Royal Society of London B : Biological Sciences*, 366(1575) :2247–2259, 2011.

[108] 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., and Gandrillon, O. 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, 2016.

[109] Vecchio, D. D. and Murray, R. M. Biomolecular feedback systems. 91(2) :220–221, 2016.

[110] Leduc, M., Gautier, E.-F., Guillemin, A., Broussard, C., Salnot, V., Lacombe, C., Gandrillon, O., Guillonneau, F., and Mayeux, P. Deep proteomic analysis of chicken erythropoiesis. *bioRxiv*, 2018.

[111] Svensson, V., Vento-Tormo, R., and Teichmann, S. A. Exponential scaling of single-cell rna-seq in the past decade. *Nature protocols*, 13(4) :599, 2018.

[112] Moris, N., Edri, S., Seyres, D., Kulkarni, R., Domingues, A. F., Balayo, T., Frontini, M., and Pina, C. 2018.

[113] Guillemin, A., Duchesne, R., Crauste, F., Gonin-Giraud, S., and Gandrillon, O. Drugs modulating stochastic gene expression affect the erythroid differentiation process. *bioRxiv*, 2018.

[114] Kupiec, J. A probabilistic theory for cell differentiation, embryonic mortality and dna c-value paradox. *Speculations in Science and Technology*, 6(5) :471–478, 1983.

[115] Braun, E. The unforeseen challenge : from genotype-to-phenotype in cell populations. *Reports on Progress in Physics*, 78(3) :036602, 2015.

[116] Petersen, M. B. K., Azad, A., Ingvorsen, C., Hess, K., Hansson, M., Grapin-Botton, A., and Honore, C. Single-cell gene expression analysis of a human esc model of pancreatic endocrine development reveals different paths to beta-cell differentiation. *Stem Cell Reports*, 9(4) :1246–1261, 2017.

[117] Jeong, H., Tombor, B., Albert, R., Oltvai, Z. N., and Barabási, A.-L. The large-scale organization of metabolic networks. *Nature*, 407(6804) :651, 2000.

[118] Jang, S., Choubey, S., Furchtgott, L., Zou, L. N., Doyle, A., Menon, V., Loew, E. B., Krostag, A. R., Martinez, R. A., Madisen, L., Levi, B. P., and Ramanathan, S. Dynamics of embryonic stem cell differentiation inferred from single-cell transcriptomics show a series of transitions through discrete cell states. *Elife*, 6, 2017.

[119] Deng, Y. and Du, W. The kantorovich metric in computer science : A brief survey. *Electronic Notes in Theoretical Computer Science*, 253(3) :73–82, 2009.

[120] Herbach, U. *Modélisation stochastique de l'expression des gènes et inférence de réseaux de régulation*. PhD thesis, Université Claude Bernard Lyon 1, 2018.

[121] Kieffer-Kwon, K. R., Nimura, K., Rao, S. S. P., Xu, J., Jung, S., Pekowska, A., Dose, M., Stevens, E., Mathe, E., Dong, P., Huang, S. C., Ricci, M. A., Baranello, L., Zheng, Y., Tomassoni Ardori, F., Resch, W., Stavreva, D., Nelson, S., McAndrew, M., Casellas, A., Finn, E., Gregory, C., St Hilaire, B. G., Johnson, S. M., Dubois, W., Cosma, M. P., Batchelor, E., Levens, D., Phair, R. D., Misteli, T., Tessarollo, L., Hager, G., Lakadamyali, M., Liu, Z., Floer, M., Shroff, H., Aiden, E. L., and Casellas, R. Myc regulates chromatin decompaction and nuclear architecture during b cell activation. *Mol Cell*, 67(4) :566–578 e10, 2017.

[122] Ugarte, F., Sousae, R., Cinquin, B., Martin, E. W., Krietsch, J., Sanchez, G., Inman, M., Tsang, H., Warr, M., Passegue, E., Larabell, C. A., and Forsberg, E. C. Progressive chromatin condensation and h3k9 methylation regulate the differentiation of embryonic and hematopoietic stem cells. *Stem Cell Reports*, 5(5) :728–740, 2015.

[123] Klein, A. M., Mazutis, L., Akartuna, I., Tallapragada, N., Veres, A., Li, V., Peshkin, L., Weitz, D. A., and Kirschner, M. W. Droplet barcoding for single-cell transcriptomics applied to embryonic stem cells. *Cell*, 161(5) :1187–1201, 2015.

[124] Rosenberg, A. B., Roco, C. M., Muscat, R. A., Kuchina, A., Sample, P., Yao, Z., Graybuck, L. T., Peeler, D. J., Mukherjee, S., Chen, W., Pun, S. H., Sellers, D. L., Tasic, B., and Seelig, G. Single-cell profiling of the developing mouse brain and spinal cord with split-pool barcoding. *Science*, 360(6385) :176–182, 2018.

[125] Sugimura, R., Jha, D. K., Han, A., Soria-Valles, C., da Rocha, E. L., Lu, Y.-F., Goettel, J. A., Serrao, E., Rowe, R. G., Malleshaiah, M., Wong, I., Sousa, P., Zhu, T. N., Ditadi, A., Keller, G., Engelman, A. N., Snapper, S. B., Doulatov, S., and Daley, G. Q. Haematopoietic stem and progenitor cells from human pluripotent stem cells. *Nature*, 545 :432.

[126] Lis, R., Karrasch, C. C., Poulos, M. G., Kunar, B., Redmond, D., Duran, J. G. B., Badwe, C. R., Schachterle, W., Ginsberg, M., Xiang, J., Tabrizi, A. R., Shido, K., Rosenwaks, Z., Elemento, O., Speck, N. A., Butler, J. M., Scandura, J. M., and Rafii, S. Conversion of adult endothelium to immunocompetent haematopoietic stem cells. *Nature*, 545 :439.

[127] Kreutz, C. and Timmer, J. Systems biology : experimental design. *FEBS J*, 276(4) :923–42, 2009.

[128] Ud-Dean, S. M. and Gunawan, R. Optimal design of gene knockout experiments for gene regulatory network inference. *Bioinformatics*, 32(6) :875–83, 2016.