



HAL
open science

La cellule, un calculateur analogique chimique

François Fages, Franck Molina

► **To cite this version:**

François Fages, Franck Molina. La cellule, un calculateur analogique chimique. Approches symboliques de la modélisation et de l'analyse des systèmes biologiques., ISTE group, pp.255-273, 2022, 9781789480290. hal-02518419

HAL Id: hal-02518419

<https://inria.hal.science/hal-02518419>

Submitted on 25 Mar 2020

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

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

La cellule, un calculateur analogique chimique

François FAGES

● Inria Saclay Ile-de-France, Palaiseau, France

Franck MOLINA

● Lab. Sys2diag CNRS-Alcediag, Montpellier, France

1.1. Introduction

La cellule est l'unité biologique structurelle et fonctionnelle fondamentale de tous les organismes vivants connus. C'est aussi la plus petite unité capable de se

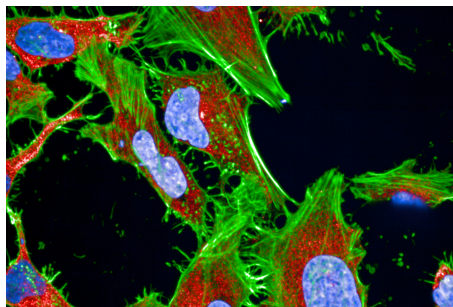


Figure 1.1. *Image de cellules (@Servier).*

reproduire de façon autonome dans un environnement propice. Pour cela, les cellules traitent l'information des signaux chimiques qu'elles reçoivent de leur environnement, en opérant des fonctions de filtrage du bruit, d'amplification du signal, et souvent de conversion analogique-digitale pour la production de réponses en tout ou rien. Ensuite, les cellules se trouvent prendre des décisions qui vont impacter leur destin cellulaire : décisions de réplication (entrée dans le cycle de division cellulaire), d'apoptose (suicide cellulaire), de migration, de différenciation (changement de type cellulaire par changement d'expression des gènes), de changement du métabolisme, etc. Enfin, les cellules contrôlent et régulent l'exécution de ces processus complexes : comme par exemple la bonne progression dans le cycle de division cellulaire, la réparation de l'ADN endommagé, ou les échanges transmembranaires qui permettent de réguler certains paramètres vitaux comme le volume ou le pH de la cellule.

Bref, pour un informaticien les cellules calculent, mais de façon particulièrement déroutante, avec des réactions chimiques dans une joyeuse soupe, certes compartimentée, d'ADN, d'ARN, de protéines et de petites molécules, dont on voudrait comprendre les principes fondamentaux de calcul et de programmation, indépendamment de la question de l'absence d'intentionnalité.

Bien évidemment, ces principes de calcul et programmation biochimiques paraissent extrêmement différents de ceux de nos ordinateurs. D'une part, les calculs d'une cellule s'effectuent le plus souvent de manière visiblement analogique, sans codage digital de l'information, avec des activations graduelles de protéines pour lesquelles le temps que prennent les réactions compte. D'autre part, l'ensemble des programmes qu'une cellule peut exécuter, qui est codé implicitement dans son ADN par la combinatoire des protéines que la cellule peut synthétiser, a été acquis par l'évolution Darwinienne, ce qui n'est pour le moins pas une forme classique de programmation de nos ordinateurs, mais qui toutefois se pratique bel et bien en Intelligence Artificielle avec d'énormes succès, par exemple pour la programmation des réseaux de neurones artificiels par un très long mécanisme d'apprentissage automatique de type évolutionnaire sur de grandes quantités de données, qui précède leur déploiement, une fois programmés de cette manière, pour résoudre ensuite des problèmes pratiques difficiles de façon routinière (lecture de chèques, reconnaissance faciale, etc.).

La thèse de Church-Turing énonce cependant qu'il n'y a qu'une seule notion universelle de calcul mécaniste, qui donc, d'une façon ou d'une autre, doit permettre de reconcilier le traitement de l'information en informatique et en biologie cellulaire.

Dans ce chapitre, nous montrons comment la preuve de Turing-complétude des réseaux de réactions chimiques continus sur un nombre fini d'espèces moléculaires, propose les bases d'une telle réconciliation, sans construction artificielle, avec des capacités à la fois d'*analyse* des programmes chimiques naturels dans les *cellules*

vivantes, et de *synthèse* de programmes chimiques artificiels dans des *vésicules non vivantes*.

Ces résultats ouvrent un vaste champ de recherche d'informatique fondamentale sur l'*informatique analogique chimique de la cellule* avec des applications nouvelles en santé et environnement.

1.2. Réseaux de réactions chimiques

Les Réseaux de Réactions Chimiques (RRC) sont un formalisme de base en chimie et biologie pour décrire les systèmes complexes d'interactions moléculaires. D'un point de vue purement formel, on peut se restreindre à des réactions élémentaires à au plus deux réactants, sans nécessairement conservation de la masse ni d'autres quantités physiques. Nous allons donc nous intéresser aux schémas de réaction suivants :

- 1) $\emptyset \rightarrow c$
- 2) $a \rightarrow \emptyset$
- 3) $a \rightarrow c$
- 4) $a \rightarrow c + d$
- 5) $a + b \rightarrow \emptyset$
- 6) $a + b \rightarrow c$
- 7) $a + b \rightarrow c + d$
- 8) $a + b \rightarrow c + d + e$

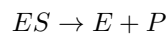
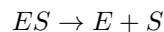
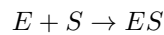
dans lesquels les lettres peuvent dénoter la même espèce moléculaire, ce qui change la nature de la réaction dans ce cas.

Par exemple, le schéma formel 4, $a \rightarrow c + d$, qui s'interprète naturellement comme une réaction de dissociation de a en c et d , représente une réaction de synthèse de d catalysée par a si $a = c$. De même, le schéma 6, $a + b \rightarrow c$, représente soit une réaction d'association de a et b pour former c , soit une réaction de dégradation active de b catalysée par a si $a = c$. Le schéma 7 avec $a = c$ représente les réactions enzymatiques typiques de transformation de b en d catalysées par une enzyme a , comme par exemple les réactions de phosphorylation (resp. déphosphorylation) d'une protéine par une enzyme kinase (resp. phosphatase) a .

Avant même de parler de cinétique chimique, ces schémas de réactions constituent en quelque sorte les instructions dont on dispose dans le «langage de programmation du vivant». Bien sûr, les schémas de réactions de synthèse de type 1, ou de dégradation de type 2 ou 5, ne peuvent pas exister sous cette forme dans la nature. Ils doivent être compris comme des réactions formelles abstraites réalisables

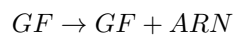
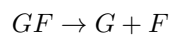
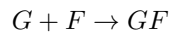
avec des espèces cachées qui ne sont pas représentées dans le système, soit parce qu'elles n'interagissent pas avec les variables du système auquel on s'intéresse, soit parce qu'elles demeurent en excès à concentration quasi-constante. Par exemple, la réaction formelle d'annihilation de type 5 peut être implémentée simplement par une réaction de complexation formant un complexe stable ab . De même, le schéma 1 de synthèse formelle peut s'interpréter comme une réaction concrète d'activation d'une espèce cachée inactive c' .

Les réactions enzymatiques sont à la base des processus de commande et régulation de la cellule, notamment avec l'emploi des enzymes kinases et phosphatases qui sont capables de modifier les protéines par ajouts et suppressions de groupes phosphates sur certains de ses acides aminés, ce qui change leurs capacités d'interaction. Une réaction enzymatique dans laquelle une enzyme E se lie à un substrat S de façon réversible pour former un complexe ES capable de transformer le substrat en produit P , peut donc se décrire par le RRC formé des 3 réactions de type 6, 4, 4 suivantes :

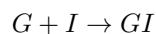


ou bien sous forme simplifiée dite de Michaelis-Menten avec une seule réaction de type 7 catalysée par E : $E + S \rightarrow E + P$.

De même, la synthèse de l'ARN d'un gène G auquel s'est lié un facteur de promotion F , peut se modéliser par le RRC suivant :



La seconde réaction de synthèse de l'ARN peut être décrite de façon plus détaillée si besoin, en ajoutant par exemple les réactions de liaison de l'ARN polymérase qui opère la transcription de la suite codée des nucléotides du gène en ARN. L'action d'un inhibiteur I à ce processus peut se modéliser par exemple en ajoutant une réaction de liaison compétitive :



qui réprime l'expression du gène puisqu'aucune réaction ne s'applique alors.

Particulièrement intrigante est la structure d'un RRC de signalisation que l'on retrouve dans toutes les cellules eucaryotes (c.-à-d. à noyau), des levures aux mammifères, et en plusieurs copies dans une même cellule avec différentes espèces moléculaires pour différents récepteurs transmembranaires : les réseaux MAPK

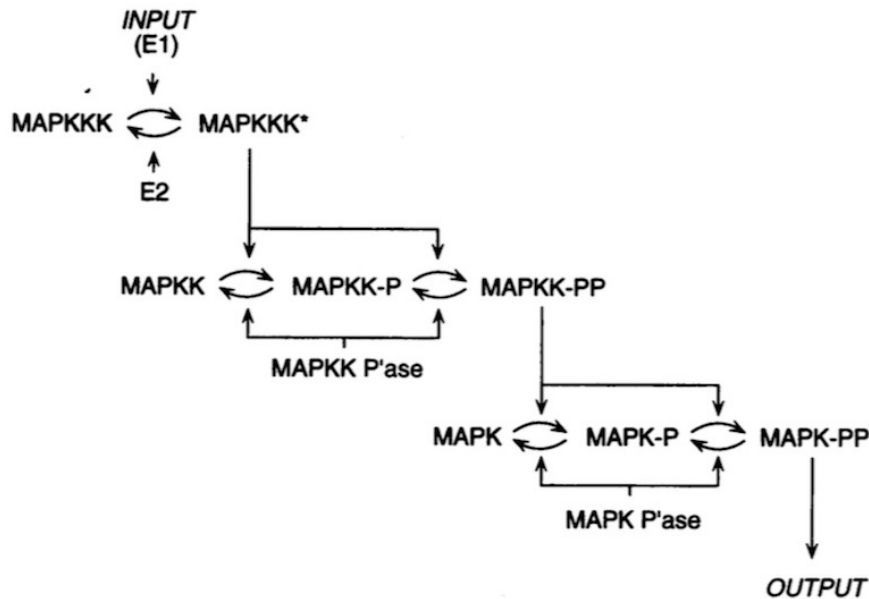


Figure 1.2. (Huang and Ferrell 1996) Structure des réseaux de signalisation MAPK universellement présents et en plusieurs copies dans les cellules eucaryotes. L'entrée E_1 est un récepteur transmembranaire activé par un ligand extérieur comme une hormone par exemple. La sortie MAPK-PP est une protéine doublement phosphorylée capable de migrer dans le noyau et de modifier l'expression de certains gènes. Les espèces intermédiaires MAPKK et MAPKKK sont des kinases de kinases qui phosphorylent les autres kinases en cascade une fois elles-mêmes phosphorylées.

(Mitogen Activated Protein Kinases). Cette structure de RRC schématisée dans la figure 1.2, comprend trois étages en cascade de double phosphorylation, qui transmettent le signal chimique de l'entrée (un récepteur transmembranaire activé par un ligand) jusqu'à sa sortie (une protéine capable de migrer dans le noyau et modifier l'expression des gènes).

Mais quelle est donc la fonction de cette structure de RRC si particulière et pourtant omniprésente dans les cellules ?

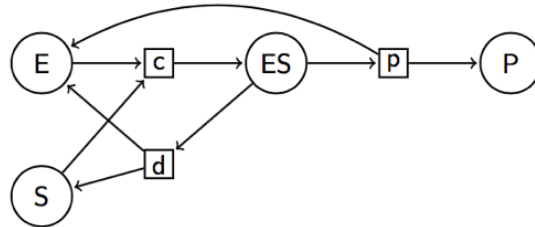


Figure 1.3. Structure de graphe biparti espèces-réactions (i.e. réseau de Petri noté avec des ronds pour les places représentant les espèces et des carrés pour les transitions représentant les réactions) du RRC enzymatique de Michaelis-Menten.

1.3. Dynamiques discrètes et calcul digital

Dans un premier temps, il est naturel pour un informaticien de chercher à interpréter les RRC précédents dans des formalismes discrets de calcul digital. En particulier, le formalisme graphique des réseaux de Petri, introduit en 1962 par Carl Adam Petri pour raisonner sur les systèmes à événements discrets (systèmes manufacturiers, réseaux de transport, et aujourd’hui *workflows*, diagrammes UML, protocoles de communication, programmes à exécution concurrente, etc.), mais inventé par lui-même paraît-il à l’âge de 13 ans en 1939 pour décrire les processus chimiques, s’applique ici directement.

Par exemple, la figure 1.3 montre la structure du réseau de Petri du schéma de réaction enzymatique précédent, $E + R \rightarrow ES$, $ES \rightarrow E + S$, $ES \rightarrow E + P$, sous la forme d’un graphe biparti, avec des nœuds “places” pour les espèces moléculaires, et des nœuds “transitions” pour les trois réactions de complexation c , décomplexation d , et production p .

Dans la dynamique discrète d’un réseau de Petri, l’état du système est défini par le nombre de jetons dans chaque place, c.-à-d. ici le nombre (ou niveau) de molécules de chaque espèce moléculaire (0 ou 1 dans le cas d’un gène, un nombre entier dans les autres cas). Une transition (réaction) est applicable si toutes les places prédécesseurs (réactants) ont des jetons, et les changements d’état s’opèrent en déplaçant les jetons de la réaction choisie dans un ordre arbitraire. Tous les ordres de déclenchement possibles des transitions sont considérés. On obtient ainsi une dynamique non-déterministe asynchrone (une seule réaction se déclenche à la fois), et on peut s’intéresser à certaines propriétés d’invariance dans un réseau donné. Par exemple dans le réseau précédent, il existe deux *invariants de place* (c.-à-d. ensembles de places où le nombre de jetons total reste toujours le même quelles que soient les transitions opérées) :

$$|E| + |ES| = \text{constante}$$

$$|S| + |ES| + |P| = \text{constante}$$

Suivant ces principes, sur le réseau de signalisation MAPK comme sur des RRC plus complexes de commande du cycle de division des cellules de mammifères (500 espèces moléculaires, 800 réactions), on a pu appliquer des algorithmes de preuve de circuits par *model-checking* formel (Chabrier-Rivier *et al.* 2004) pour vérifier l'ensemble des comportements qualitatifs éventuellement possibles exprimés en logique temporelle (accessibilité, points de passages obligés, oscillations, etc.), ou bien calculer des invariants du réseau de Petri : lois de conservations, flux extrêmes, points d'accumulation etc. (Nabli *et al.* 2016).

Mais cela ne suffit pas du tout à comprendre le choix d'une structure aussi particulière que le réseau MAPK par exemple pour traiter les signaux d'une cellule.

1.4. Dynamiques continues et calcul analogique

Les interprétations discrètes fournissent des approximations supérieures des comportements qualitatifs possibles (Fages and Soliman 2008), mais dans la réalité biologique le temps que prennent les réactions chimiques compte, et le sous-ensemble des comportements effectivement observés en biologie ne peut pas s'expliquer sans prendre en compte les vitesses des réactions.

Dans les RRC décomposés en réactions élémentaires, on peut se restreindre à des cinétiques régies par la loi d'action de masse qui stipule que le nombre de réactions par unité de temps est proportionnel aux quantités de réactants. Le taux d'une réaction à deux réactants, A, B, est alors un monôme de la forme $k.A.B$, où le facteur de proportionnalité k caractérise l'affinité de A et B à réagir lors d'une collision.

Ces fonctions cinétiques donnent la vitesse des réactions en chaque état du système. Cela fournit immédiatement une interprétation continue des RRC dans laquelle les états sont définis par les concentrations (à valeurs réelles) de chaque espèce moléculaire, et les transitions continues sont régies par le système d'équations différentielles ordinaires qui, pour chaque espèce, somme les fonctions cinétiques des réactions qui produisent l'espèce, et soustrait celles des réactions qui la consomment. Dans l'exemple du RRC de Michaelis-Menten, on obtient

$$dS/dt = -k_1.E.S + k_2.ES$$

$$dE/dt = -k_1.E.S + (k_2 + k_3).ES$$

$$dES/dt = k_1.E.S - (k_2 + k_3).ES$$

$$dP/dt = k_3.ES$$

On note que les deux invariants de place du réseau de Petri de la figure 1.3 donnent bien deux invariants algébriques structurels (linéaires et indépendants des valeurs

des paramètres cinétiques) de ce système d'équations différentielles, qui permettent d'éliminer deux variables : $dE/dt + dES/dt = 0$, $dS/dt + dES/dt + dP/dt = 0$, d'où en tout point de temps t , $E(t) + ES(t) = E(0) + ES(0)$ et $S(t) + ES(t) + P(t) = S(0) + ES(0) + P(0)$. C'est une propriété générale qui lie les invariants du réseau de Petri et l'interprétation différentielle des RRC.

Les autres fonctions cinétiques considérées notamment pour les réactions enzymatiques, comme les cinétiques de Michaelis-Menten de la forme $\frac{v.E.S}{k + S}$ pour la réaction $E + S \rightarrow E + P$, soit

$$dP/dt = -dS/dt = \frac{v.E.S}{k + S},$$

ou bien de Hill de la forme $\frac{v.E.S^n}{k^n + S^n}$ dans le cas d'une chaîne de réactions enzymatiques coopératives,

proviennent de réductions de RRC élémentaires avec la loi d'action de masse, sous des hypothèses par exemple d'état quasi-stationnaire de certaines variables, ou de quasi-équilibre de certaines réactions, notamment lorsque $E \ll S$ (Segel 1984).

Notons que pour les petits nombres de molécules (comme les gènes qui sont en exemplaire unique dans une cellule), l'interprétation différentielle considère des activations fractionnaires qui n'ont pas grand sens. Une autre façon plus réaliste d'interpréter un RRC est alors de lui associer une chaîne de Markov à temps continu, c.-à-d. un automate non-déterministe semblable au réseau de Petri précédent, mais probabilisé à la fois sur le choix de la prochaine réaction et sur le temps (continu) de celle-ci. Dans cette interprétation stochastique, les fonctions cinétiques définissent, en chaque état du système, une propension pour chaque réaction à se déclencher, qui après normalisation sur toutes les réactions applicables dans cet état, fournit une probabilité de réaction et la loi de distribution du temps de déclenchement de la prochaine réaction (Gillespie 1977, 2009).

La figure 1.4 montre une simulation différentielle du RRC de signalisation MAPK de la figure 1.2, pour une concentration donnée de récepteurs activés en entrée. S'agissant d'assez grands nombres de molécules, les simulations stochastiques donneraient ici des courbes similaires à la simulation continue. On remarque l'amplification du signal aux niveaux 2 et 3 par rapport au niveau 1 (parce que les protéines KK et K des niveaux inférieurs sont simplement en plus grand nombre par trois ordres de grandeur que la protéine KKK du premier niveau), ainsi que la saturation rapide de la sortie Kpp qui intervient avant celle de l'étage intermédiaire et illustre l'ultrasensibilité de la réponse.

Mais plus intéressante est la figure 1.5, qui décrit le niveau de concentration des sorties de chacun des trois étages de la cascade MAPK après stabilisation, en faisant

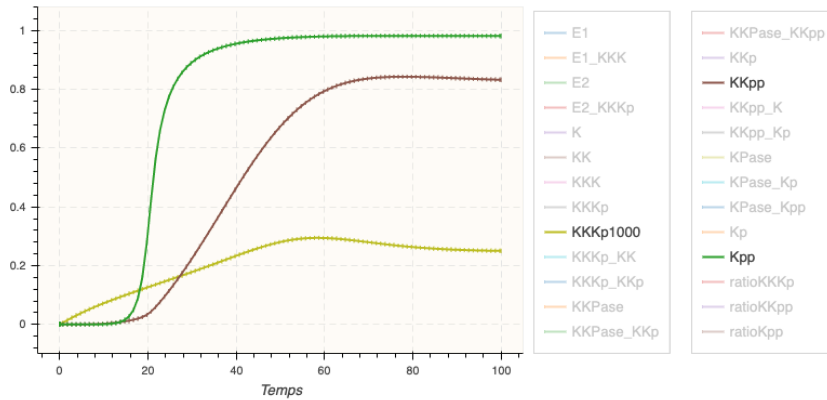


Figure 1.4. Evolution au cours du temps des niveaux de concentration des sorties du RRC de signalisation cellulaire MAPK de la figure 1.2 pour chacun des 3 étages ($KKKp$ multiplié par 10^3 , $KKpp$, et Kpp), obtenue par simulation des équations différentielles du modèle MAPK de Huang and Ferrell (1996). Les espèces moléculaires intermédiaires en gris dans la légende ne sont pas tracées. Cette simulation montre la forte amplification du signal de l'étage 1 aux sorties des étages 2 et 3, la saturation de la sortie de l'étage 3 obtenue avant celle de l'étage intermédiaire 2, et l'ultrasensibilité de la réponse.

varier graduellement le niveau de l'entrée (concentration de récepteurs activés par un ligand). Cette figure, appelée *diagramme dose-réponse* par les biologistes, *diagramme de bifurcation* par les mathématiciens, et *spécification fonctionnelle* par les informaticiens, montre ici l'amplification du signal dès le second étage, et surtout une réponse en sigmoïde raide au troisième étage de sortie de la cascade, presque en tout ou rien autour d'une certaine valeur de seuil de l'entrée. Cette figure justifie donc la structure particulière, si bien conservée entre les espèces, du réseau MAPK à 3 étages de phosphorylation, comme une façon d'implémenter un *convertisseur analogique-digital* en biochimie.

Dès lors que la concentration d'une molécule en sortie d'un RRC se stabilise pour chaque valeur de concentration des entrées, on peut dire que *le RRC calcule une fonction réelle d'entrée-sortie*. On ne connaît pas d'expression analytique de la fonction d'entrée-sortie $Kpp(E_1)$ précise du réseau MAPK, mais la courbe de réponse en sortie peut être assez bien approximée par une fonction de Hill d'ordre 5 (Huang and Ferrell 1996), c.-à-d.

$$Kpp(E_1) \sim v \cdot \frac{E_1^5}{k + E_1^5}$$

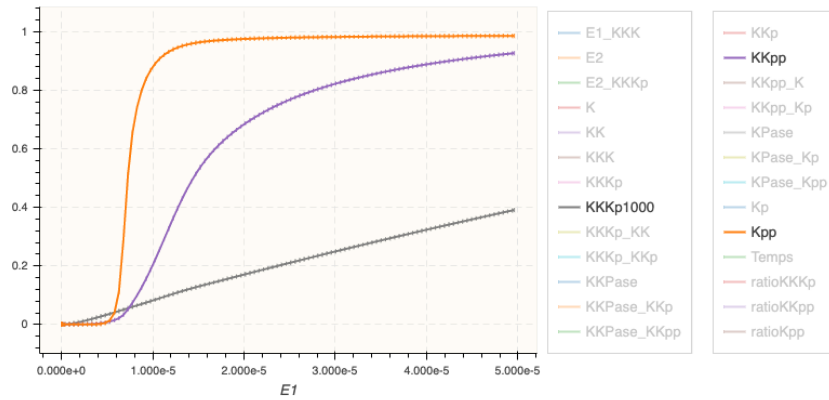


Figure 1.5. Diagramme dose-réponse montrant le graphe de la fonction d'entrée-sortie "calculée" par le réseau MAPK sur les sorties de chacun des trois étages MAPK ($KKKp$ multiplié par 10^3 , KKp , Kpp) en fonction de la concentration de l'entrée E_1 . La réponse en "tout ou rien" obtenue à l'étage 3 de la sortie contrairement aux réponses graduelles obtenues aux deux premiers étages montre la fonction de convertisseur analogique-digital de ce réseau, qui explique la structure particulière à 3 étages de phosphorylation des réseaux MAPK naturels.

où v est un facteur d'amplification du signal (directement lié à la concentration initiale de la kinase K non phosphorylée du troisième étage de sortie) et k la valeur du seuil qui résulte des paramètres cinétiques des réactions de phosphorylation/déphosphorylation.

Dans le contexte du calcul chimique, la précision des calculs n'est bien sûr pas un objectif, mais un moyen mathématique pour définir le type de fonction calculée par un RRC. Cette vue amène alors naturellement plusieurs questions :

- Peut-on implémenter de façon théoriquement exacte la fonction $f(x) = \frac{x^5}{1 + x^5}$ par un RRC fini ?
- Comment se compare ce RRC au circuit MAPK ?
- Pourrait-on le faire fonctionner dans des vésicules artificielles de synthèse ?
- Et peut-on en théorie implémenter toute fonction réelle calculable par un RRC fini ?

1.5. Turing-complétude des réseaux de réactions chimiques continus

La thèse de Church-Turing énonce qu'il n'y a qu'une seule notion de calcul effectif sur les structures de données discrètes, et par extension sur les nombres réels en précision arbitraire (donnés par des suites convergentes de nombres rationnels). Elle s'appuie sur la preuve donnée par Alan Turing en 1936, d'équivalence du pouvoir d'expression du λ -calcul de Church qui établit le fondement des mathématiques constructives, et des machines de Turing qui modélisent les procédés de calcul mécanistes, comme ceux que réalisent les ordinateurs aujourd'hui pour l'exécution de tout programme. Du point de la calculabilité, le choix d'un langage de programmation n'a donc pas d'importance, dès lors qu'il a la puissance d'expression du λ -calcul ou d'une machine de Turing.

Il est bien connu (mais non trivial) que les réseaux de Petri ne sont pas Turing-complets. Il s'ensuit que dans cette interprétation discrète, les RRC (sur un nombre fini d'espèces moléculaires) ne sont pas Turing-complets. Ils le deviennent néanmoins si on autorise des réactions de polymérisation, qui fournissent alors un codage de l'information dans des polymères non bornés, c.-à-d. sur un ensemble infini d'espèces moléculaires assez artificiel (Qian *et al.* 2011, Cook *et al.* 2009).

Il est aussi bien connu aussi que dans leur interprétation par équations différentielles, les RRC continus, tout comme les réseaux de neurones, fournissent un moyen de calcul universel pour une notion de calcul non-uniforme : pour toute fonction réelle calculable, pour toute donnée en entrée, il existe un RRC continu qui calcule le résultat de la fonction sur la donnée en entrée (Helmfelt *et al.* 1991, Magnasco 1997). Dans cette approche, le codage digital de l'entrée nécessite cependant d'augmenter la taille du RRC pour les différentes entrées. Or la complétude de Turing demande d'invertir les quantificateurs, c.-à-d. de montrer que pour toute fonction calculable il existe un RRC continu fini qui calcule le résultat quelle que soit la donnée en entrée.

C'est ce que nous avons montré en nous appuyant sur des travaux antérieurs de Bournez *et al.* (2007) sur la Turing-complétude des systèmes d'équations différentielles polynomiales (PIVP), et sur l'inférence de RRC à partir d'équations différentielles (Fages *et al.* 2015). En montrant d'une part, que l'on peut se restreindre à des polynômes de degré au plus 2 (pour les réactions élémentaires à au plus deux réactants), et d'autre part, à des calculs dans le quadrant positif pour la restriction aux valeurs positives de concentration (résolue en codant une variable réelle x par la différence $x_p - x_m$ des concentrations de deux variables positives associées à x (Oishi and Klavins 2011)), on déduit :

Théorème. (Fages *et al.* 2017) Toute fonction réelle calculable peut être implémentée par un RRC continu sur un nombre fini d'espèces moléculaires.

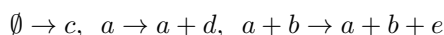
Un aspect particulièrement intéressant de ce résultat est qu'il ne s'appuie pas sur une construction artificielle qui nous éloignerait des RRC naturels. Dans cette approche,

- les concentrations moléculaires sont le support analogique de l'information ;
- l'ensemble des programmes d'une cellule est défini implicitement par l'ensemble des parties de l'ensemble des réactions possibles avec les protéines codées dans l'ADN de la cellule ;
- le programme analogique en cours d'exécution d'une cellule est défini par l'état courant d'expression des gènes,
- les changements de programme d'une cellule résultent des changements d'expression des gènes, que l'on peut assimiler à un méta-programme digital.

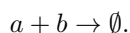
Ainsi dans cette vue d'une cellule, les données sont codées de façon analogique dans les concentrations moléculaires, et les programmes analogiques de façon digitale dans l'ADN.

La preuve du théorème précédent montre aussi que l'on peut se restreindre plus précisément à des réactions de synthèse et dégradation de type 1 et 5, et des réactions catalysées de synthèse de type 4 et 8 (avec $a = c$, $b = d$) :

Théorème. (Fages *et al.* 2017) Une fonction réelle est calculable si et seulement si elle est calculable par un RRC fini utilisant des réactions de synthèse à au plus deux catalyseurs des trois formes suivantes



et des réactions de dégradation par annihilation de la forme



Comme dit plus haut, de telles réactions de synthèse et annihilation n'existent pas dans la nature, mais doivent être comprises comme des schémas de réactions réalisables avec des espèces cachées ; typiquement, pour le schéma d'annihilation, le complexe ab dans une réaction de complexation ; et pour les différents schémas de synthèse, la forme inactive, c' , d' ou e' , de l'espèce moléculaire produite.

Il est aussi intéressant de noter qu'une réaction d'annihilation, $a + b \rightarrow \emptyset$, calcule la différence de concentration $|a - b|$ en l'affectant au réactant initialement en plus grande quantité, tandis que l'autre passe à 0. C'est le principe que nous avons utilisé plus haut pour normaliser la représentation d'une variable réelle x codée par deux variables positives : x_p pour les valeurs positives et x_m pour les valeurs négatives.

On peut alors aussi remarquer avec intérêt que pour cette même raison de calcul de différence entre a et b , une telle réaction d'annihilation constitue à elle seule un

système de commande intégrale d'homéostasie, pour toute variable régulée par a et b de façon anthétique (Briat *et al.* 2016).

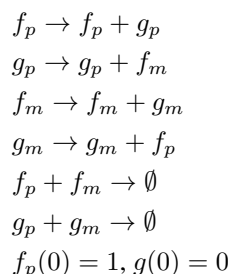
1.6. Compilateur chimique de fonctions calculables

Les résultats théoriques précédents fournissent une méthode de conception automatisée de RRC pour toute fonction réelle calculable, présentée comme solution d'un système d'équations différentielles polynomial avec des valeurs initiales également polynomiales en fonction des entrées. Ce compilateur est implémenté suivant ces principes dans le logiciel BIOCHAM-4¹ de modélisation, analyse et synthèse de RRC, ainsi qu'avec certaines variantes dans le système CRN++ (Vasic *et al.* 2018).

Par exemple, l'oscillateur le plus simple, la fonction du temps cosinus, est solution de l'équation différentielle du second ordre $f'' = -f$ avec $f(0) = 1$, soit au premier ordre

$$df/dt = -g, dg/dt = f, f(0) = 1, g(0) = 0$$

qui produit par compilation le RRC suivant à 4 espèces moléculaires, f_p, f_m, g_p, g_m pour les valeurs positives et négatives de f (cosinus) et g (sinus) :



Toutes ces réactions sont interprétées par défaut avec pour cinétiques des lois d'action de masse, de paramètre 1 pour les premières, et une constante plus grande de plusieurs ordres de grandeurs pour les réactions de normalisation des valeurs positives et négatives par annihilation. La figure 1.6 montre une simulation numérique des équations différentielles associées à ce RRC. La figure 1.7 présente une simulation stochastique de ce même RRC, qui montre *a posteriori* que l'interprétation plus réaliste de ce réseau par une chaîne de Markov à temps continu, introduit du bruit dans les amplitudes des oscillations, mais pas dans la période.

La fonction de Hill d'ordre 5, $h(x) = \frac{x^5}{1+x^5}$, qui approxime la fonction d'entrée-sortie du RRC de signalisation MAPK de la figure 1.5 (Huang and Ferrell 1996), peut être implémentée de façon exacte avec le RRC non élémentaire suivant :

1. <http://lifeware.inria.fr/biocham4>

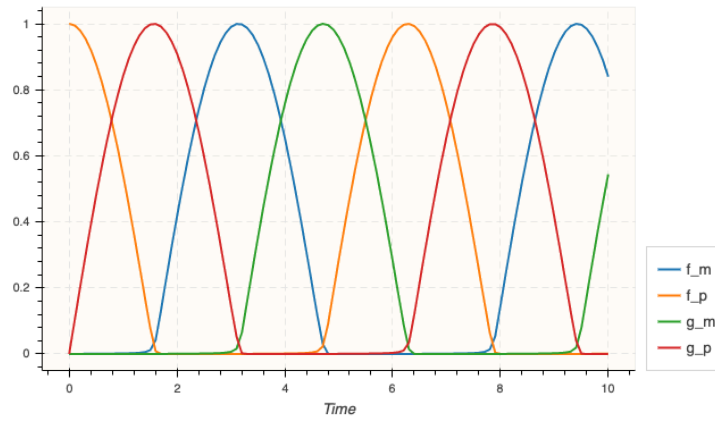


Figure 1.6. Simulations différentielle du RRC synthétisé à partir de la définition par équations différentielles de la fonction du temps cosinus. Les valeurs positives et négatives de cosinus sont représentées respectivement par les concentrations des espèces f_p et f_m dont les pics alternent. Les variables introduites g_p et g_m représentent sinus.

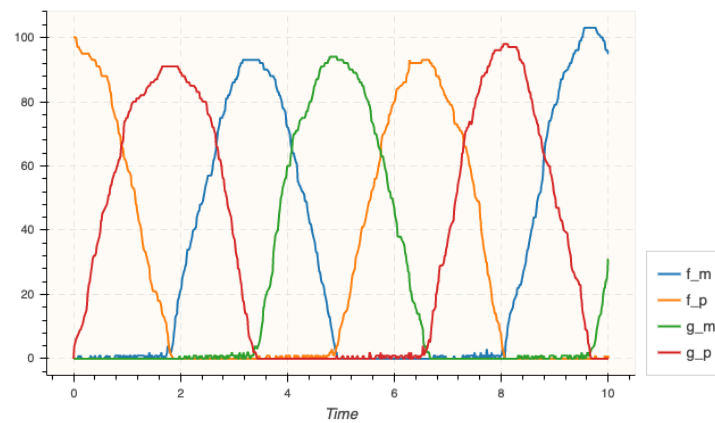
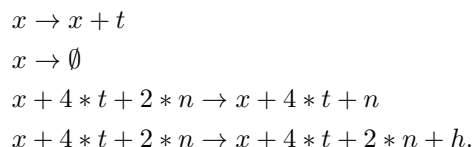


Figure 1.7. Simulations stochastique du RRC synthétisé pour la fonction du temps cosinus, montrant la résistance de ce réseau au bruit intrinsèque vis à vis de la période, et une plus grande variabilité des amplitudes.



avec des cinétiques d'action de masse de paramètre 1 pour les deux premières réactions et 5 pour les deux dernières. On vérifie aisément par dérivation formelle que ces deux dernières réactions implémentent la fonction du temps $h(t)$, mais avec x ajouté comme catalyseur, ce qui a pour effet de faire converger le système vers le temps $t = x(0)$ avec $x = 0$ par les deux premières réactions, et calcule donc $h(x)$. C'est un principe général de transformation d'un PIVP qui calcule une fonction du temps en un PIVP qui calcule une fonction d'entrée sortie, qui consiste à arrêter le temps du premier sur la valeur de l'entrée du second (Pouly 2015).

Dans le cas d'une compilation vers un réseau de réactions élémentaires, une seconde transformation de binomialisation est nécessaire se restreindre à des réactions à au plus deux réactants. Dans l'exemple de la fonction Hill d'ordre 5, cela conduit à un RRC de 7 espèces et 12 réactions, donc de complexité comparable au réseau MAPK mais sans lien évident de structure avec lui. Ce RRC synthétique offre donc une solution alternative au RRC naturel MAPK pour une fonction similaire, avec éventuellement de moins bonnes propriétés de robustesse ou d'évolvabilité qu'il serait intéressant d'analyser.

L'implémentation chimique de telles fonctions sigmoïdes est une étape importante pour se ramener au calcul digital avec des valeurs de concentrations dans l'intervalle $[0,1]$ prises proches soit de 0, soit de 1. En suite de quoi, l'implémentation des fonctions Booléennes peut se faire de multiples façons dans des RRC analogiques, par exemple en utilisant les opérations arithmétiques

- de multiplication (minimum) pour la conjonction $c = a \wedge b$, avec la réaction $a + b \rightarrow c$;
- d'addition (maximum) pour la disjonction $d = a \vee b$, avec les réactions $a \rightarrow d$, $b \rightarrow d$, $a + b \rightarrow \emptyset$;
- et de soustraction pour la négation $e = \neg a$, avec la réaction $a + e \rightarrow \emptyset$ et $e(0) = 1$.

1.7. Programmation chimique de vésicules non-vivantes

Le compilateur chimique précédent permet de concevoir des RRC abstraits pour implémenter une fonction donnée. L'implémentation pratique de tels réseaux nécessite cependant de trouver des réactions existant réellement sur les entrées et les sorties d'intérêt.

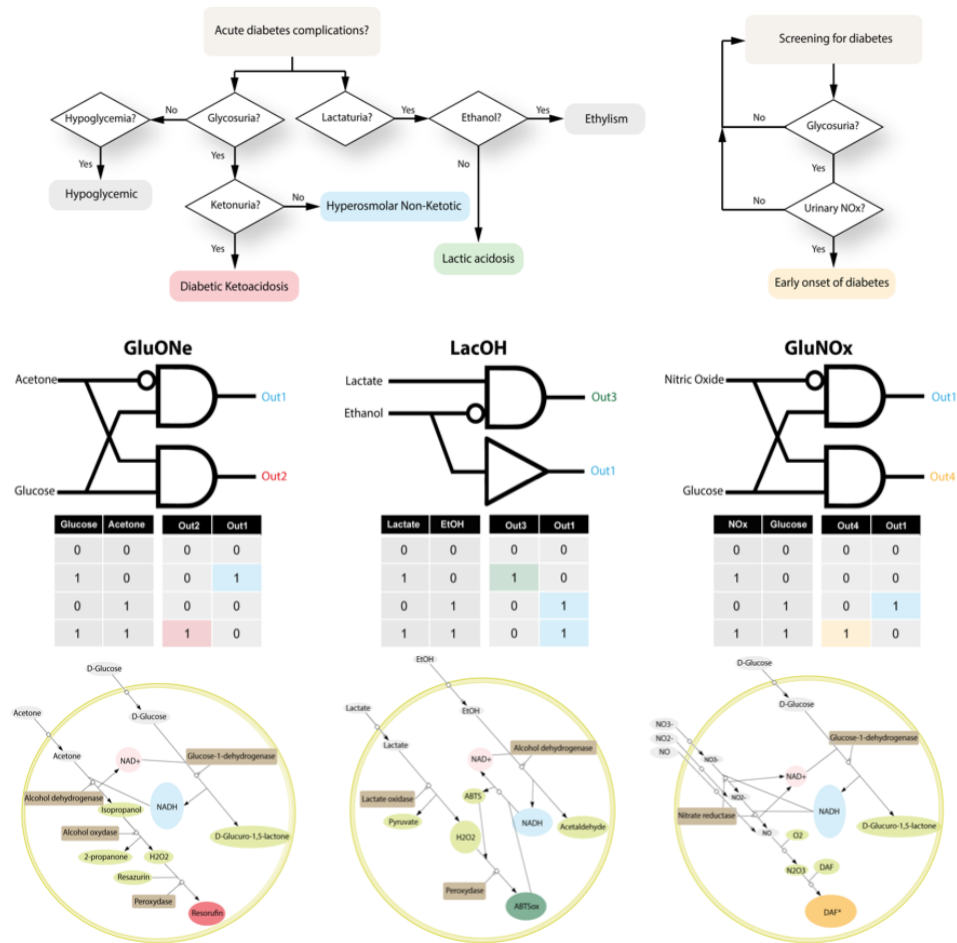


Figure 1.8. Algorithmes de diagnostic de différents types de comas (diabétique de deux types, éthylique, ou lactique), circuits logiques correspondants (testant la présence ou l'absence de glucose, acétone, éthanol, lactate), et graphes des RRCs correspondants (produisant en sortie une coloration ou fluorescence par accumulation de NADH, Résorufine, ABTS et DAF-2), implémentés dans des vésicules artificielles qui ont été testées dans des urines et du sang de patient (Courbet et al. 2018).

Bien qu'allant à l'encontre de la majeure partie des recherches en biologie synthétique encore aujourd'hui, particulièrement intéressantes sont les réactions

enzymatiques qui ne font pas intervenir l'ADN et tout l'appareillage nécessaire de transcription et traduction, car elles permettent de se passer de l'utilisation de cellules vivantes comme "machine hôte", et permettent d'envisager la réalisation de RRC autonomes et indépendants, encapsulés dans des vésicules non-vivantes entièrement synthétiques.

Il s'agit alors de trouver dans les bases d'enzymes, comme par exemple la base Brenda², des réactions concrètes capables d'implémenter les schémas précédents pour les entrées et les sorties d'intérêt, en introduisant les espèces intermédiaires nécessaires.

C'est ce que réalise le laboratoire CNRS-Alcediag Sys2diag à Montpellier, pour la fabrication de biocapteurs conçus à partir de circuits logiques de diagnostic. Ces biocapteurs sont alors produits sous la forme de vésicules artificielles créées par des circuits micro-fluidiques. La figure 1.8 montre les étapes de conception de tels RRC pour un algorithme simple de diagnostic de différentes formes de comas, implémenté biochimiquement dans des vésicules synthétiques sans ADN (Courbet *et al.* 2018).

Dans cet exemple, la robustesse du RRC vis à vis de propriétés de la fonction calculée, exprimées en logique temporelle quantitative (Rizk *et al.* 2009), a été optimisée dans Biocham en déterminant les quantités optimales de chaque enzyme à mettre dans la vésicule initialement. Ensuite, si les entrées auxquelles le programme réagit sont présentes, le programme s'exécute en consommant son énergie et s'éteint, comme le font certaines de nos cellules qui ont perdu par différenciation leur noyau et leur appareil de traduction (ex. cellules de peau, globules rouges, etc.), mais continuent néanmoins de fonctionner pendant plusieurs dizaines de jours.

Par exemple, le circuit de gauche dans la figure 1.8 détecte la présence de glucose et d'acétone (sortie colorée rouge par production de resorufine), ainsi que la présence de glucose sans acétone (sortie colorée bleue par accumulation de NADH). Ce circuit a été implémenté avec un RRC constitué de 26 réactions élémentaires et 25 espèces moléculaires (schématisé en bas à gauche dans la figure), et testé sur des urines et du sang de patients pour valider son fonctionnement et sa stabilité pendant plusieurs semaines.

1.8. 10^{14} calculateurs analogiques en réseau

Notre corps est formé de l'ordre de 3.10^{13} cellules humaines et 4.10^{13} bactéries (Sender *et al.* 2016), c.-à-d. dans notre miroir informatique, de l'ordre de 10^{14} calculateurs analogiques, dont la moitié partagent le même méta-programme (nos gènes), et l'autre moitié (ceux de notre microbiote aux méta-programmes spécialisés

2. <https://www.brenda-enzymes.org/>

plus limités mais variés et changeants) proviennent de notre environnement, sous le contrôle vigilant de notre système immunitaire.

Une des leçons des 80 ans d'histoire de l'informatique était que seul le calcul digital était capable de passer à l'échelle des grands systèmes, contrairement au calcul analogique. La biologie cellulaire semble nous indiquer le contraire dans le cas du calcul biochimique. La biologie synthétique nous fournit des technologies pour expérimenter le concept de programmation biochimique, notamment sans ADN pour la synthèse de vésicules aux applications nouvelles en santé et environnement. L'informatique théorique nous fournit des outils de conception de programmes chimiques analogiques. L'intelligence artificielle nous fournit d'autres outils complémentaires de conception de réseaux chimiques par évolution artificielle (Degrand *et al.* 2019) qui peuvent nous rapprocher de l'évolution biologique, vue comme un gigantesque algorithme d'apprentissage (Valiant 2013).

Toutes les conditions semblent donc réunies pour développer maintenant une nouvelle branche de l'informatique : la *bio-informatique analogique cellulaire*.

1.9. Bibliographie

- Bournez, O., Campagnolo, M. L., Graça, D. S., Hainry, E. (2007), Polynomial differential equations compute all real computable functions on computable compact intervals, *Journal of Complexity*, 23(3), 317–335.
- Briat, C., Gupta, A., Khammash, M. (2016), Antithetic integral feedback ensures robust perfect adaptation in noisy biomolecular networks, *Cell Systems*, 2(1), 15–26.
- Chabrier-Rivier, N., Chiaverini, M., Danos, V., Fages, F., Schächter, V. (2004), Modeling and querying biochemical interaction networks, *Theoretical Computer Science*, 325(1), 25–44.
- Cook, M., Soloveichik, D., Winfree, E., Bruck, J. (2009), Programmability of chemical reaction networks, in A. Condon, D. Harel, J. N. Kok, A. Salomaa, E. Winfree, (eds), *Algorithmic Bioprocesses*, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 543–584.
- Courbet, A., Amar, P., Fages, F., Renard, E., Molina, F. (2018), Computer-aided biochemical programming of synthetic microreactors as diagnostic devices, *Molecular Systems Biology*, 14(4).
- Degrand, E., Hemery, M., Fages, F. (2019), On chemical reaction network design by a nested evolution algorithm, in CMSB'19 : Proceedings of the seventeenth international conference on Computational Methods in Systems Biology, Lecture Notes in Bioinformatics, Springer-Verlag.
- Fages, F., Gay, S., Soliman, S. (2015), Inferring reaction systems from ordinary differential equations, *Theoretical Computer Science*, 599, 64–78.

- Fages, F., Le Guludec, G., Bournez, O., Pouly, A. (2017), Strong Turing Completeness of Continuous Chemical Reaction Networks and Compilation of Mixed Analog-Digital Programs, in CMSB'17 : Proceedings of the fifteen international conference on Computational Methods in Systems Biology, vol. 10545 of *Lecture Notes in Computer Science*, Springer-Verlag, pp. 108–127.
- Fages, F., Soliman, S. (2008), Abstract interpretation and types for systems biology, *Theoretical Computer Science*, 403(1), 52–70.
- Gillespie, D. T. (1977), Exact stochastic simulation of coupled chemical reactions, *Journal of Physical Chemistry*, 81(25), 2340–2361.
- Gillespie, D. T. (2009), Deterministic limit of stochastic chemical kinetics, *The Journal of Physical Chemistry B*, 113(6), 1640–1644.
- Helmfelt, A., Weinberger, E. D., Ross, J. (1991), Chemical implementation of neural networks and turing machines, *PNAS*, 88, 10983–10987.
- Huang, C.-Y., Ferrell, J. E. (1996), Ultrasensitivity in the mitogen-activated protein kinase cascade, *PNAS*, 93(19), 10078–10083.
- Magnasco, M. O. (1997), Chemical kinetics is turing universal, *Physical Review Letters*, 78(6).
- Nabli, F., Martinez, T., Fages, F., Soliman, S. (2016), On enumerating minimal siphons in petri nets using CLP and SAT solvers : Theoretical and practical complexity, *Constraints*, 21(2), 251–276.
- Oishi, K., Klavins, E. (2011), Biomolecular implementation of linear i/o systems, *IET Systems Biology*, 5(4), 252–260.
- Pouly, A. (2015), Continuous models of computation : from computability to complexity, PhD thesis, Ecole Polytechnique.
- Qian, L., Soloveichik, D., Winfree, E. (2011), Efficient turing-universal computation with DNA polymers, in Proc. DNA Computing and Molecular Programming, vol. 6518 of *LNCS*, Springer-Verlag, pp. 123–140.
- Rizk, A., Batt, G., Fages, F., Soliman, S. (2009), A general computational method for robustness analysis with applications to synthetic gene networks, *Bioinformatics*, 12(25), i169–i178.
- Segel, L. A. (1984), *Modeling dynamic phenomena in molecular and cellular biology*, Cambridge University Press.
- Sender, R., Fuchs, S., Milo, R. (2016), Revised estimates for the number of human and bacteria cells in the body, *PLoS Biology*, 14(8).
- Valiant, L. (2013), *Probably Approximately Correct, Nature's Algorithms for Learning and Prospering in a Complex World*, Basic Books.
- Vasic, M., Soloveichik, D., Khurshid, S. (2018), CRN++ : Molecular programming language, in Proc. DNA Computing and Molecular Programming, vol. 11145 of *LNCS*, Springer-Verlag, pp. 1–18.