



# Recherche de modules dans le réseau MAPK

Laurent Chamoin

► **To cite this version:**

Laurent Chamoin. Recherche de modules dans le réseau MAPK. [Intership report] 2007. <inria-00185032>

**HAL Id: inria-00185032**

**<https://hal.inria.fr/inria-00185032>**

Submitted on 5 Nov 2007

**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.

# Recherche de modules dans le réseau MAPK

Laurent Chamoin, Master 2 Bio Informatique, Rennes 1

18 Juin 2007

## Table des matières

<b>1</b>	<b>Notion de modularité</b>	<b>2</b>
1.1	Module . . . . .	2
1.2	Première méthode de réduction : les échelles de temps . . . . .	3
1.3	Deuxième méthode de réduction : Circuits positifs et négatifs . . . . .	5
1.4	Troisième méthode de réduction : Systèmes monotones . . . . .	6
1.5	Objectif du stage . . . . .	6
<b>2</b>	<b>Modèle MAPK</b>	<b>6</b>
2.1	Informations existantes . . . . .	6
2.1.1	MAPK : Principales fonctionnalités . . . . .	7
2.1.2	Genesis . . . . .	8
2.1.3	modèle de Kholodenko . . . . .	8
2.1.4	modèle de Bhalla . . . . .	12
2.2	Données pour une nouvelle implémentation informatique . . . . .	14
2.2.1	Informations de Bhalla . . . . .	14
2.2.2	Informations biologiques . . . . .	14
2.3	Intérêts d'étudier MAPK . . . . .	16
<b>3</b>	<b>But du stage</b>	<b>17</b>
3.1	Gale Nikaido . . . . .	17
3.2	Travail précédent . . . . .	18
3.3	Différentes représentations . . . . .	18
3.3.1	MATLAB . . . . .	18
3.3.2	MAPLE . . . . .	19
3.4	Code général . . . . .	19
3.4.1	Représentation symbolique . . . . .	21
3.4.2	Données diverses . . . . .	21
3.5	Difficulté : Données plus importantes . . . . .	22
3.5.1	Différentes approches . . . . .	22
3.5.2	Amélioration : Gain de temps . . . . .	23
3.6	Résultats . . . . .	24
	<b>Références</b>	<b>26</b>

## Introduction

La représentation des connaissances biologiques sous la forme d'un schéma ou croquis est courante. Cependant au vu des connaissances qui s'accumulent au fil des années, la complexité devient si importante, qu'il est nécessaire d'utiliser les outils mathématiques et informatiques pour pouvoir les exploiter et ainsi en tirer des informations. Il existe à ce jour une méthode pour permettre de simplifier les réseaux ou de les rendre exploitables, les modules. Les méthodes existantes pour déterminer où se trouvent les modules au sein des réseaux, se basent actuellement sur des informations biologiques, un critère de temps et même sur des équations mathématiques. L'objectif de ce stage est d'explorer une nouvelle méthode, basée sur une propriété mathématique, en l'appliquant à un réseau de grande taille : MAPK , en voulant au final pouvoir réduire ce réseau et le rendre exploitable.

A l'origine de ce choix, il y a la collaboration entre Anne Siegel et Ovidiu Radulescu de l'équipe Symbiose à l'IRISA qui ont développé la méthode lors d'une publication précédente, et Upinder Bhalla du National Centre for Biological Sciences à Bangalore en Inde qui travaille beaucoup sur le système biologique de MAPK. Lors de ce stage, comme dans tout travail de recherche, la recherche bibliographique a été importante, et les informations qui sont nécessaires pour comprendre l'objectif final du stage sont nombreuses. Alors dans un premier temps, on va présenter les informations connues sur la notion de modules ainsi que les méthodes jusque là utilisées pour les détecter au sein d'un réseau. Ensuite on présentera le réseau biologique en lui même afin de bien repérer les difficultés qu'un tel réseau représente, ce qui permettra finalement de pouvoir présenter le travail effectué lors de ce stage, en présentant le modèle implémenté, les difficultés rencontrées et les résultats obtenus.

## 1 Notion de modularité

Comment il est dit dans l'introduction, la biologie systémique se base sur des données diverses et nombreuses pour construire des réseaux biologiques, ce qui conduit à des réseaux de grandes tailles qui ne sont pas exploitables en l'état. Nous allons voir une méthode souvent avancée pour permettre d'exploiter ces réseaux : les modules. Dans un premier temps, nous allons définir ce qui est appelé module pour ensuite s'intéresser à trois méthodes classiques trouver des modules au sein d'un réseau : les échelles de temps, l'analyse des circuits et les systèmes monotones.

### 1.1 Module

Lorsqu'on regarde les différents réseaux biologiques disponibles, on remarque rapidement que si certains, le plus souvent de petite taille, semblent aisés à comprendre, d'autres qui représentent une grande quantité de connaissances, peuvent sembler difficilement analysables instinctivement. C'est pourquoi, pour répondre à ce problème, la notion de module est apparue.

Par définition, un module est dans un réseau, un regroupement de sommets. L'explication de ce regroupement est variable. Il existe différentes méthodes qui permettent de regrouper des sommets. La première que l'on utilise est celle qui est la plus intuitive. On se sert de la biologie pour déterminer quels sommets seront regroupés sous la forme d'un module. Par exemple, en biologie si des interactions sont compartimentées, on utilisera ces compartiments naturels pour délimiter un module. On aura ainsi une cellule qui contiendra un

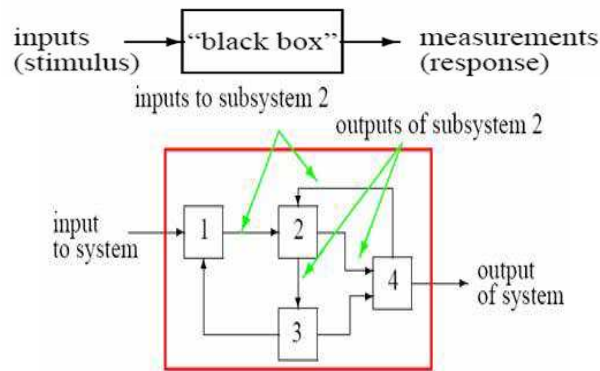


FIG. 1 – Schéma montrant le principe de la boîte noire(en haut) et schéma montrant ce qu'un module peut cacher en son sein, avec la possibilité qu'un module soit un regroupement de modules(en bas).Exemple tiré des publications de [10]

module mitochondrial, et un module pour le noyau par exemple.

D'un autre côté, le regroupement en module n'est pas toujours aussi intuitif, et on regroupera ces sommets en modules soient en se basant sur des données biologiques, autres que le compartiment naturel, ou bien suivant une application mathématique. Si avec les données biologiques, l'apport se fait d'un point de vue fonctionnel, c'est à dire chercher les circuits dans un réseau et déterminer les états d'équilibre, comme nous le verrons, en ce qui concerne les applications mathématiques, permettant de trouver des modules n'ayant pas forcément de sens biologique, leurs intérêts résident dans la nécessité de réduire les grands réseaux pour pouvoir les exploiter, c'est ce que permettent les échelles de temps et les systèmes monotones que nous allons voir.

Pour donner une image simple de ce qu'est un module, on prend celle de la boîte noire, qui accepte des informations en entrée, et en fournit d'autres en sortie.

Ce qui se passe entre les entrées et les sorties est masqué mais on sait que cette boîte noire simule le comportement de plusieurs sommets du réseau biologique qui a été ainsi simplifié avec l'aide des modules comme le montre la figure 1.

L'important lorsque l'on conçoit un module, est de savoir comment simuler le comportement. Cette réponse est fournie par une équation mathématique qui permet de transformer les paramètres pour en faire des données de sorties.

## 1.2 Première méthode de réduction : les échelles de temps

La première méthode pour la détection de modules, existe dans les faits depuis longtemps, et est utilisée par les chimistes régulièrement. Il s'agit d'exploiter les différentes échelles de temps pour 'équibrer certaines variables du système.Un exemple classique est l'obtention de l'équation de Michaelis Menten



FIG. 2 – Réaction enzymatique type, avec l'action de l'enzyme sur un substrat qui en s'associant forment un complexe qui peut soit se rompre soit donner lieu à une activité enzymatique qui transforme un substrat en produit

La justification qui permet d'utiliser une telle équation est l'utilisation des échelles de temps. En effet si on part de la partie gauche de l'équation complète d'une réaction enzymatique type, comme montré sur la figure 2, on considère que la réaction (1) et la réaction (2) se font très rapidement et donc au final la concentration en complexe Enzyme-Substrat (ES) est constante. Ainsi on écrit la formule :

$$\frac{d[ES]}{dt} = k_1 \cdot [E] \cdot [S] - k_{-1}[ES] - k_2[ES] = 0$$

où  $k_1$ ,  $k_{-1}$  et  $k_2$  correspondent respectivement aux vitesses de réactions 1, 2 et 3.

Ensuite pour déterminer la concentration en enzyme capable d'interagir avec les molécules de substrat, on va utiliser une concentration  $E_0$  regroupant l'ensemble des enzymes contenues dans le milieu, c'est à dire celles libres et celles associées dans le complexe Enzyme-Substrat. Et ainsi en réarrangeant les équations, on trouve la concentration du complexe Enzyme-Substrat. Ainsi on a,

$$[ES] = \frac{[E][S]}{K_m}$$

,où  $K_m$  équivaut à  $\frac{k_{-1}+k_2}{k_1}$ , qui nous permet d'avoir :

$$[ES] = \frac{([E_0] - [ES]) \cdot [S]}{K_m}$$

ce qui permet d'avoir ensuite

$$[ES] = [E_0] \cdot \frac{[S]}{K_m + [S]}$$

Cette concentration Enzyme-Substrat va nous servir lorsque l'on va s'intéresser à la partie droite de la réaction type, car en partant d'une équation exprimant la concentration de P en fonction de la concentration du Substrat de la réaction (3),

$$\frac{d[P]}{dt} = k_3[ES]$$

le complexe Enzyme-Substrat, on réarrange le tout pour connaître la concentration en produit exprimée en fonction de la concentration totale d'enzyme dans le milieu et la concentration en Substrat :

$$\frac{d[P]}{dt} = k_2 \cdot [E_0] \cdot \frac{[S]}{K_m + [S]}$$

Ainsi on a réussi à simplifier notre réaction type de l'action enzymatique sur un substrat pour obtenir un produit, en calculant l'équation représentant une réaction(4) plus simple : Substrat donne Produit par l'action d'Enzyme, le complexe intermédiaire [ES] ayant disparu du système..

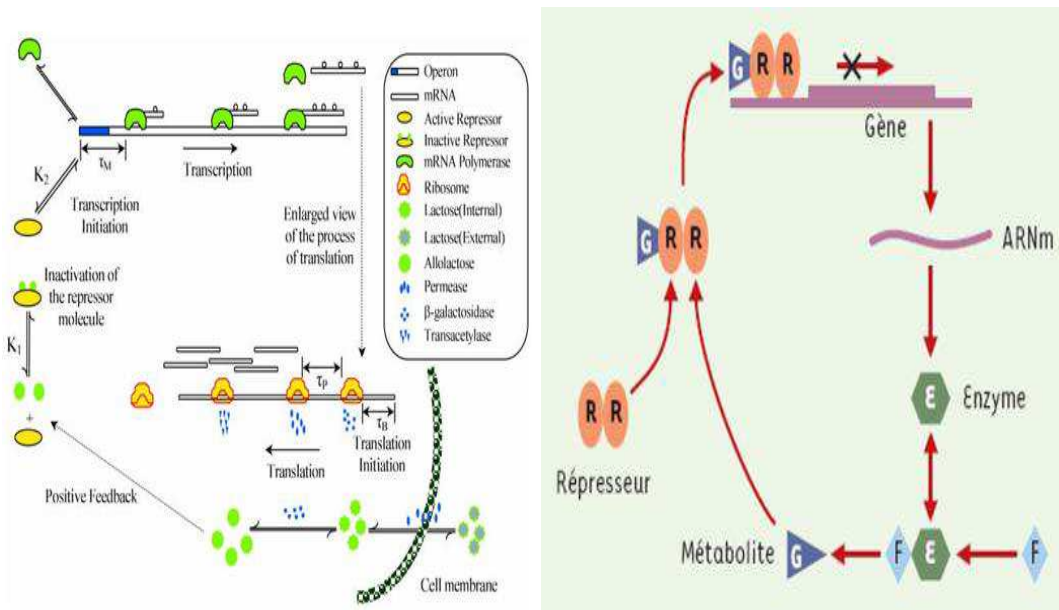


FIG. 3 – Schéma : Exemple de circuit positif( à gauche) : Opéron lactose, extrait de [12], et exemple de circuit négatif type( à droite),extrait de [4]

### 1.3 Deuxième méthode de réduction :Circuits positifs et négatifs

La deuxième méthode connue pour permettre de trouver des modules au sein d'un réseau, est basée sur l'analyse des circuits pouvant exister dans un système biologique.

**Définitions** On évoque le terme de circuit dans les réseaux lorsque qu'il existe une boucle ou un cycle entre des sommets du réseau. Le terme de positif ou négatif, correspond en fait, à l'impact qu'a ce cycle sur les concentrations des produits. On parlera de circuit positif, lorsque les produits du cycle concerné interagissent et qu'au final cela leur est favorable, c'est à dire que leur concentration augmente, comme il est montré dans la partie gauche de la figure 3. En biologique on parle de rétrocontrôle positif. A l'opposé, le circuit négatif correspond à une interaction entre les produits du cycle qui conduit à l'augmentation de la concentration d'un produit qui a un effet négatif sur la production des autres produits, on parle alors en biologie de rétrocontrôle négatif, et on peut en voir un exemple dans la partie droite de la figure 3.

Ces circuits peuvent être remplacés par un module simulant leurs comportements. Si on prend un exemple de circuit positif comme l'opéron lactose les équations différentielles existent et permettent ainsi de faire notre réduction de ce réseau. Et dans le cas d'un circuit négatif, il existe la méthode de Goodwin qui permet de simuler ce type d'interactions comme décrit par [7].

**Règle de René Thomas** Il existe une règle que l'on utilise pour trouver des circuits au sein d'un réseau. En effet en 1983, René Thomas a avancé le principe stipulant que si un système biologique admet un unique état d'équilibre, alors on en conclut que ce système contient un circuit négatif. De même, si en fonction des concentrations initiales, un système admet deux états stables, alors on en conclut que ce système possède un circuit positif.

$$x_1 \succeq x_2 \text{ et } u_1 \succeq u_2 \implies x(t, x_1, u_1) \succeq x(t, x_2, u_2) \quad \forall t \geq 0$$

FIG. 4 – Formule définissant les systèmes monotones

La règle de Thomas a été démontrée mathématiquement par Soulé,[11].

#### 1.4 Troisième méthode de réduction : Systèmes monotones

La troisième méthode qui permet de détecter les modules est celle proposée par [6] : les systèmes monotones.

Un système monotone signifie que si on prend deux paramètres de valeur différente et que l'on peut les classer l'un par rapport à l'autre, en sortie de système les résultats correspondants respecteront ce classement, comme l'explique la figure 4. Si un système est monotone, alors on en déduit qu'il vérifie un système d'Entrées-Sorties".

Les systèmes monotones sont généralement utilisés pour les réseaux avec interactions positives, mais on les utilise aussi pour les réseaux avec interactions négatives, avec une subtilité. On va utiliser une décomposition en deux systèmes monotones ce qui permet de simuler le comportement de l'interaction négative.

#### 1.5 Objectif du stage

Lors de ce stage, on va utiliser une nouvelle approche qui consiste à utiliser un théorème d'unicité d'équilibre partiel pour définir des modules permettant d'être réduit pour la théorie des fonctions implicites. Ceci sera développé en partie 3.

## 2 Modèle MAPK

Le réseau biologique qui nous a intéressé pour tester la nouvelle méthode de détection de module est celui de la Mitogen-activated Protein Kinase(MAPK)(ou Mitogen Associated Protein Kinase). L'intérêt principal de ce choix est que MAPK est un réseau très largement étudié, et que les bases de données contiennent énormément d'informations dessus, ce qui en fait un choix intéressant pour développer un outil fiable que l'on pourrait réutiliser pour d'autres modèles moins connus. Nous allons tout d'abord présenter les informations connues sur MAPK puis s'intéresser à deux modèles en particulier celui de Boris Kholodenko et ,particulièrement,celui de Upinder Bhalla.

### 2.1 Informations existantes

MAPK est un système biologique très étudié, et les informations disponibles sont nombreuses, nous allons ainsi pouvoir nous intéresser aux informations biologiques sur ce réseau, puis analyser les implémentations informatiques, avec le logiciel Genesis, et mathématiques, avec les modèles de Kholodenko et Bhalla.

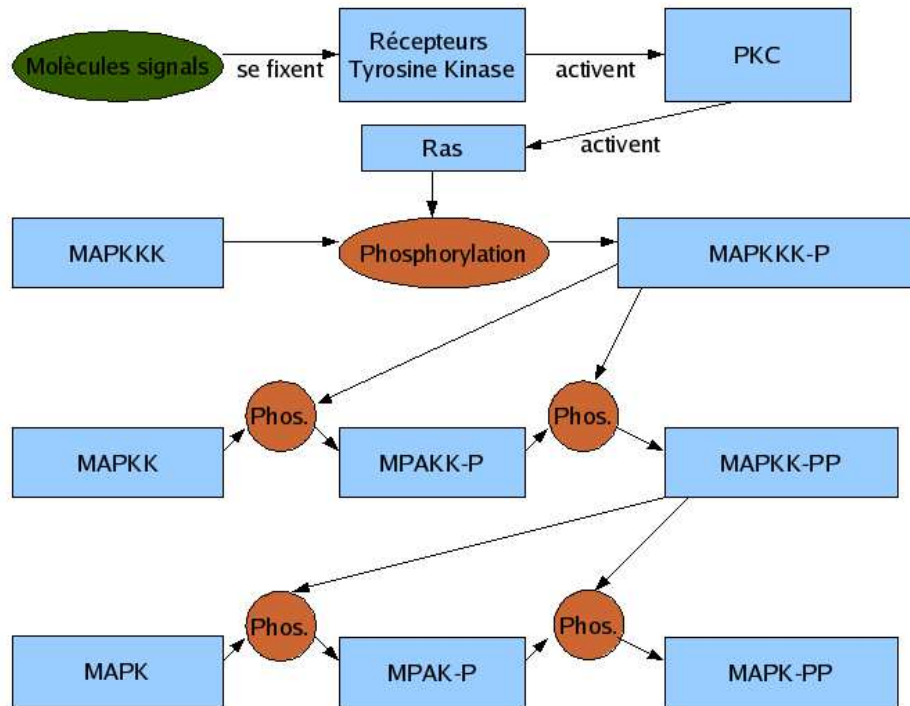


FIG. 5 – schéma : Schéma explicatif de la transmission du signal lors de la croissance cellulaire

### 2.1.1 MAPK : Principales fonctionnalités

Nous allons maintenant voir les informations biologiques connues sur le réseau MAPK, et finalement comprendre l'intérêt de vouloir l'étudier.

Un modèle biologique qui permet de retrouver des circuits positifs et négatifs est celui proposé pour la cascade de la protéine MAPK. Cette cascade a un rôle de transduction du signal dans les cellules eucaryotes. Elle est ainsi présente dans les cellules d'organismes allant de la levure aux mammifères. Plus précisément, la cascade MAPK, comme décrite par [2] sert de relais entre des stimuli extérieurs à la cellule depuis la membrane plasmique et les cibles situées dans le cytoplasme. Ceci lui permet de jouer un rôle chez les cellules des mammifères dans la différenciation, la division cellulaire, la croissance cellulaire et la réponse à un stress.

Par exemple, dans le cadre de la croissance cellulaire, représentée sur la figure 5, la molécule signal se fixe sur les récepteurs de la tyrosine kinase, ce qui leur permet d'activer par l'intermédiaire d'autres molécules la molécule PKC. PKC permet l'activation de la molécule Ras, qui est chargée de phosphoryler la première molécule de la cascade c-Raf (ou MAPKKK). Une fois cette molécule phosphorylée, elle peut phosphoryler à son tour en deux temps la molécule MEK (ou MAPKK), qui elle-même une fois doublement phosphorylée permet de rendre active en deux temps la molécule MAPK.

Autour de ce modèle biologique, on va se concentrer sur deux modèles, qui, malgré une compatibilité au niveau de la cascade, diffèrent au niveau de la complexité de représentation des rétrocontrôles.



au terme de la transmission MAPK est doublement phosphorylée donc activée et peut ainsi servir de relais au signal pour les molécules cibles se trouvant dans le cytoplasme.

### 2.1.2 Genesis

Genesis est un logiciel de simulation du comportement des concentrations dans ces deux modèles. Il s'agit d'un logiciel libre d'accès qui permet de représenter les réseaux biologiques, de définir les concentrations et les constantes de réactions dans le but final de pouvoir étudier le comportement des concentrations des différents produits représentés. Genesis s'appuie sur un modèle d'équations différentielles explicites.

La définition d'un modèle mathématique, est l'expression de la concentration des variables en s'appuyant sur la matrice stoechiométrique et une matrice de Flux, un exemple est donné lorsque l'on traitera du modèle de Kholodenko.

Bhalla propose via une base de données, présentée par [9], un ensemble de modèles concernant MAPK. Deux nous intéressent particulièrement présentés figure 6. D'un point de vue visuel, si celui de Kholodenko reste suffisamment compréhensible, la complexité du modèle de Bhalla fait qu'une telle représentation est seulement intéressante pour les résultats fournis par le logiciel et en aucun cas pour visualiser le modèle dans son ensemble. Il faut aussi noter que les modèles proposés sont implémentés le plus souvent par Bhalla, donc si on reprend le cas du modèle proposé faisant référence à la publication de Kholodenko, le modèle peut différer du modèle originel de l'auteur.

Bien que visuellement ce logiciel ne permette pas de comprendre réellement la structure des réseaux de taille importante, il permet de suivre l'évolution des concentrations de produits que l'on sélectionne. Ceci permet notamment de visualiser l'impact d'une augmentation subite, à la volée, voir figure 7, d'une concentration donnée sur les autres concentrations, comme le montre la figure.

Ainsi cet outil a pour principale utilisation la simulation et la visualisation du comportement du réseau biologique du point de vue des concentrations.

Un autre point essentiel, c'est que chaque réaction décrite dans le modèle sous Genesis est reliée par des liens bibliographiques permettant de vérifier l'existence et la validité du modèle présenté.

### 2.1.3 modèle de Kholodenko

Nous allons ainsi s'intéresser à un premier modèle qui a été proposé par Boris Kholodenko, voir figure 8.

Ce système biologique permet d'étudier le comportement oscillatoire, comme présenté par [5] des concentrations des molécules. Le rétrocontrôle proposé est une interaction directe entre la molécule MAPK doublement phosphorylée (ou MAPK-PP) et les molécules Ras qui déclenchent la réaction en cascade.

Cette interaction consiste en une réactivation des molécules Ras qui, lors de la phosphorylation de la molécule c-Raf se sont désactivées; ceci conduit, à terme, à un appauvrissement des molécules Ras actives et donc avec la réaction 2 du modèle, les MAPKKK en se déphosphorylant et en s'accumulant empêchaient toute activité dans la cascade. Ceci

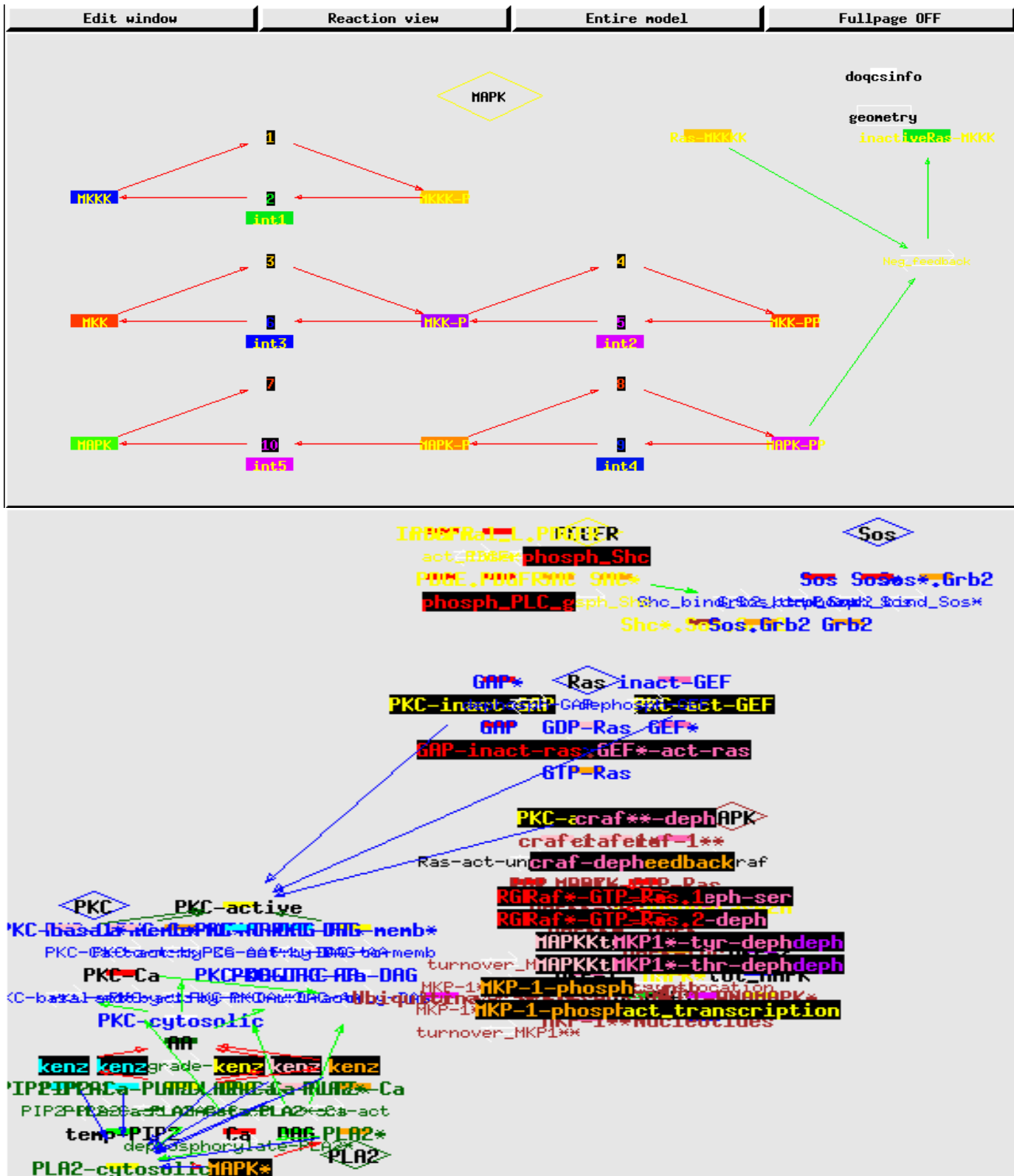


FIG. 6 – Représentation du modèle de Kholodenko par le logiciel Genesis. La cascade est facilement identifiable, et sur la partie droite on voit la représentation de l'interaction entre MAPK-PP et Ras-inactif(En haut). Et la représentation du modèle de Bhalla par le logiciel Genesis(en bas).

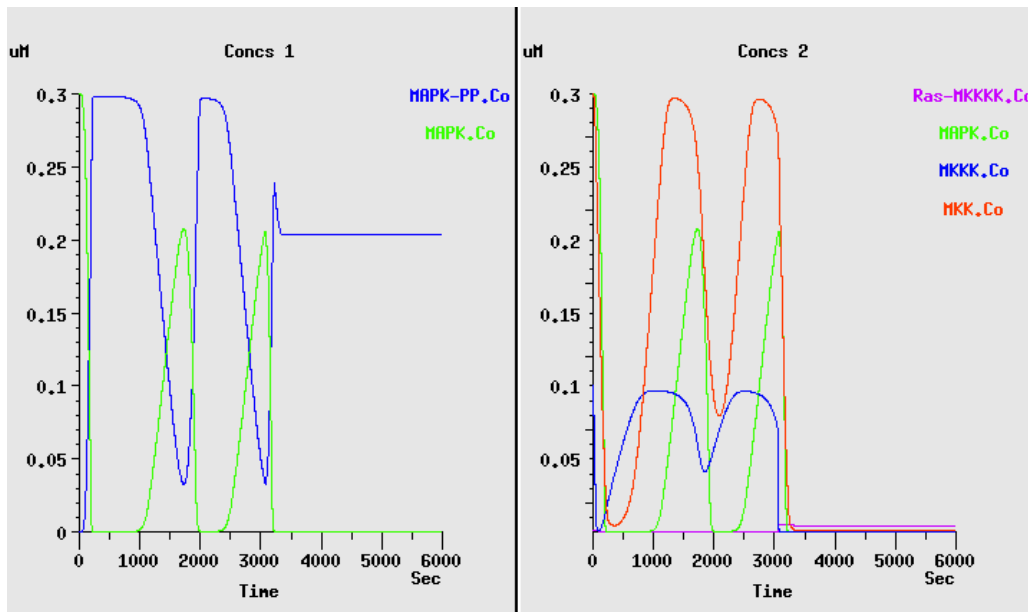


FIG. 7 – Exemple de possibilité de genesis : on rajoute en cours d'exécution (environ 3500 sec après le début de la simulation) une forte dose de Ras active, ce qui a pour effet de bloquer les molécules au stade activé.

conduit, une fois toutes les molécules de la cascade déphosphorylées, le modèle à se retrouver en situation initiale dans sa partie cascade. Du coup, avec ce rétrocontrôle positif de la part de MAPK-PP, il y a toujours des molécules Ras activées et donc les différentes phosphorylations se passant dans la cascade peuvent avoir lieu.

Dans les faits, en étudiant les concentrations des molécules, on observe le comportement oscillatoire recherché, voir figure ???. Ce comportement est cependant déterminé par la concentration initiale en Ras activé. Une trop faible quantité conduit à un état d'équilibre bas, alors qu'au contraire si on sature en Ras activé, toutes les molécules de la cascade restent dans leur état activé et on n'observe aucune oscillation.

Cependant, en s'intéressant aux équations de réactions proposées par Boris Kholodenko, on remarque que Ras-MK444 n'est pas directement pris en compte dans ces équations, mais on va se baser uniquement sur la concentration de MAPK-PP.

Il est important de noter un point particulier. Jusqu'à présent, nous avons décrit, le rétrocontrôle de MAPK-PP sur Ras comme étant positif, et si on s'intéresse aux graphiques fournis par le modèle de Kholodenko implémenté par Bhalla dans le logiciel Genesis, cela est vrai. Cependant, en reprenant la publication originelle sur ce modèle, on remarque que Kholodenko décrit ce rétrocontrôle comme étant négatif. Ceci se voit dans la première équation de réaction. Selon lui, MAPK-PP inhibe la réaction qui phosphoryle MAPK444, ainsi, plus la concentration en MAPK-PP augmente, moins de MAPK444 se fait phosphoryler, et du fait des réactions inverses, les concentrations en molécules phosphorylées décroissent. Ceci créant bien un phénomène d'oscillation.

La publication de Kholodenko, fournit les informations nécessaires pour la simulation. En effet en ayant les informations sur les flux, voir figure 10 et sur les équations différentielles, voir figure 11, on possède les informations nécessaires pour la création du modèle mathématique

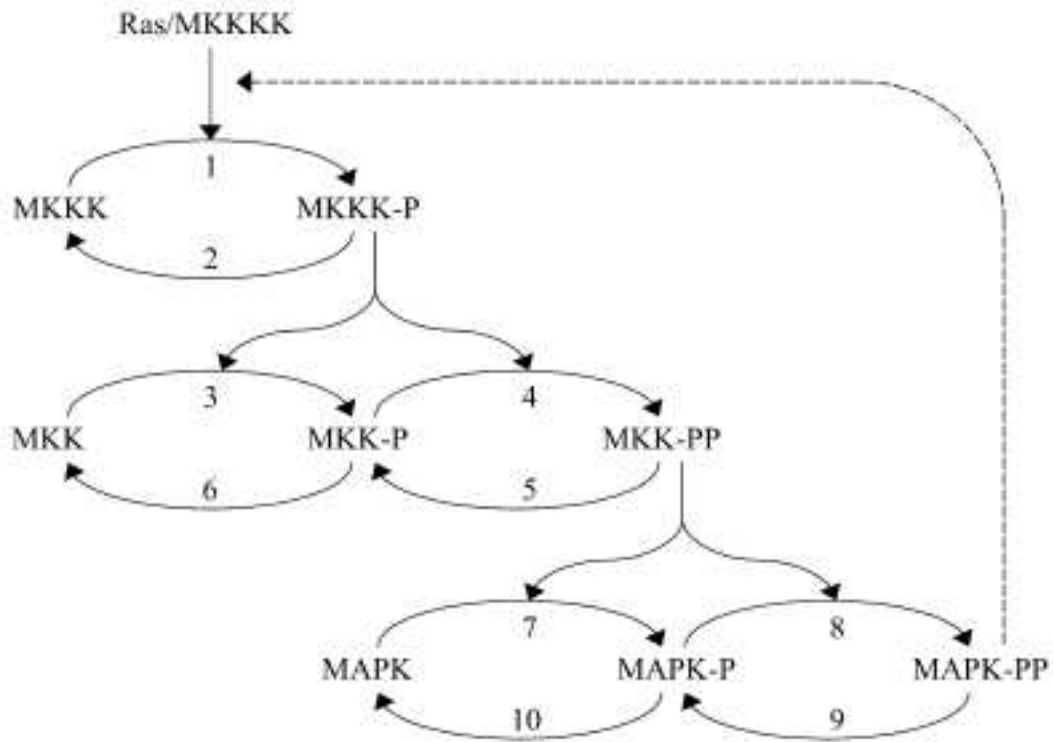


FIG. 8 – Modèle de Kholodenko. Seule la partie cascade est réellement représentée, le rétrocontrôle positif quantà lui est simplement symbolisé par la flèche en traits discontinus.

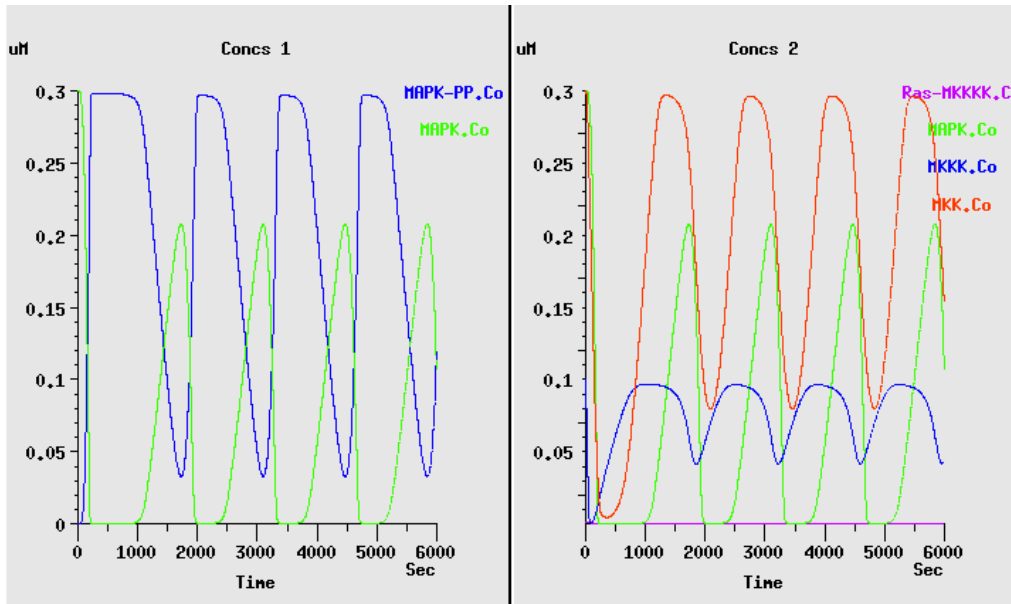


FIG. 9 – Graphique montrant l'évolution des concentrations MAPK inactive et active (à gauche) et des concentrations des molécules présentes au début de la simulation (à droite)

	Equation de réactions	Valeurs des paramètres
F1=	$V_1.[MKKK]/((1 + ([MAPK - PP]/K_1)^n).[K_1 + [MKKK]])$	$V_1 = 2.5; n = 9; K_1 = 10$
F2=	$V_2.[MKKK - P]/(K_2 + [MKKK - P])$	$V_2 = 0.25; K_2 = 8$
F3=	$k_3.[MKKK - P].[MKK]/(K_3 + [MKK])$	$k_3 = 0.025; K_3 = 15$
F4=	$k_4.[MKKK - P].[MKK - P]/(K_4 + [MKK - P])$	$k_4 = 0.025; K_4 = 15$
F5=	$V_5.[MKK - PP]/(K_5 + [MKK - PP])$	$V_5 = 0.75; K_5 = 15$
F6=	$V_6.[MKK - P]/(K_6 + [MKK - P])$	$V_6 = 0.75; K_6 = 15$
F7=	$K_7.[MKK - PP].[MAPK]/(K_7 + [MPAK])$	$k_7 = 0.025; K_7 = 15$
F8=	$k_8.[MKK - PP].[MAPK - P]/(K_8 + [MAPK - P])$	$k_8 = 0.025; K_8 = 15$
F9=	$V_9.[MAPK6PP]/(K_9 + [MAPK - PP])$	$V_9 = 0.5; K_9 = 15$
F10=	$V_{10}.[MAPK - P]/(K_{10} + [MAPK - P])$	$V_{10} = 0.5; K_{10} = 15$

FIG. 10 – Tables des flux du modèle de Kholodenko, on remarque bien que la première réaction est inversement proportionnelle à la concentration en MAPK-PP, ce qui conduit à un rétrocontrôle négatif.

de la réaction.

L'hypothèse retenue pour expliquer cette différence est que Kholodenko s'est principalement intéressé au phénomène d'oscillation dans la cascade et a juste voulu trouver un phénomène le déclenchant. De son côté, Bhalla s'est appuyé sur ses propres informations pour expliquer ces oscillations. Mais au final, ce que l'on souhaitait observer est bien présent, mais force est de constater que l'origine des oscillations est difficile à comprendre.

### 2.1.4 modèle de Bhalla

Un second modèle, proposé par Bhalla, voir figure 12, introduit deux rétrocontrôles sur la cascade, comme le montre [2]. Un premier, positif, permet par l'intermédiaire de cPlA2 et de son produit, l'acide arachidonique, de stimuler PKC, ce qui comme on l'a vu précédemment permet de lancer la cascade, mais cette fois sans stimuli extérieur. Un second, négatif, qui par l'intermédiaire du gène de la phosphatase de MAPK, appelée MKP-gène, dont l'activation se fait par MAPK-PP, déphosphoryle (désactive) la molécule MAPK. De même pour les deux autres molécules de la cascade, c-Raf et MEK sont, elles déphosphorylées par l'action de la protéine phosphatase 2A (PP2A), une phosphatase spécifique de la Ser-Thr.

En faisant un parallèle avec le modèle de Kholodenko, on remarque que le rétrocontrôle négatif et l'inhibiteur de réaction sont bien présents. En effet, les réactions 9 et 10, correspondent à l'action de MKP et les réactions 2, 5 et 6 à l'action de PP2A. De même, le rétrocontrôle positif du modèle de Kholodenko est aussi très détaillé dans le modèle de Bhalla.

Un autre point important qui différencie ces deux modèles, c'est la présence du stimulus dans le modèle de Bhalla. Dans le modèle de Kholodenko, le parcours du signal jusqu'à la molécule Ras est bien décrite mais non représentée. Dans le cas du modèle de Bhalla, le signal est sous deux formes, DAG qui active PKC et PDGF qui active par l'intermédiaire de SHC, grb2 et Sos directement Ras.

Le modèle qui va être étudié par la suite sera une version de Bhalla décrite dans une autre publication. Elle est dans l'ensemble identique à celle que l'on vient de présenter mais diffère par l'absence de l'activation de PDGF mais aussi par une plus grande précision dans les

$$\begin{aligned} \frac{d[MKKK]}{dt} &= v_2 - v_1 \\ \frac{d[MKKK-P]}{dt} &= v_1 - v_2 \\ \frac{d[MKK]}{dt} &= v_6 - v_3 \\ \frac{d[MKK-P]}{dt} &= v_3 + v_5 - v_4 - v_6 \\ \frac{d[MKK-PP]}{dt} &= v_4 - v_5 \\ \frac{d[MAPK]}{dt} &= v_{10} - v_7 \\ \frac{d[MAPK-P]}{dt} &= v_7 + v_9 - v_8 - v_{10} \\ \frac{d[MAPK-PP]}{dt} &= v_8 - v_9 \end{aligned}$$

FIG. 11 – Systèmes des équations différentielles

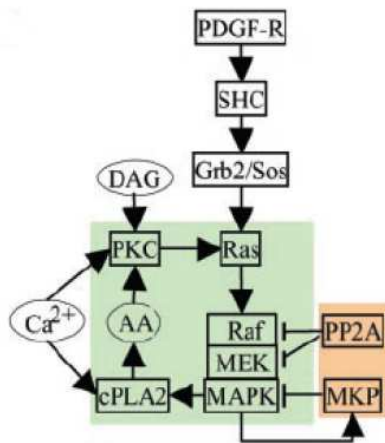


FIG. 12 – Modèle de Bhalla, on observe bien le circuit positif sur la partie gauche et le circuit négatif en bas à droite du schéma

Réactant 1	Réactant 2	Réactant 3	Produit 1	Produit 2	Produit 3
cRaf*	GTP-Ras		cRaf*-GTP-Ras		
cRaf	GTP-Ras		cRaf-GTP-Ras		
PKC-cytosolic	Ca		PKC-Ca		
DAG	PKC-Ca		PKC-Ca-DAG		
PKC-Ca			PKC-Ca-memb		
PKC-ca-DAG			PKC-DAG-memb*		
PKC-Ca	AA		PKC-Ca-AA*		
PKC-DAG-AA			PKC-DAG-AA*		
PKC-cytosolic			PKC-basal*		
AA	PKC-cytosolic		PKC-AA*		
PKC-cytosolic	DAG		PKC-DAG		
PKC-DAG	AA		PKC-DAG-AA		
MKP-1			Ubiquination		
MKP-1-gene			MKP-1		
MKP-1**			Ubiquination		
MKP-1-ser359*			Ubiquination		
MKP-1-ser359*			MKP-1		
MKP-1**			MKP-1-ser359*		
MAPK*			Nuclear-MAPK*		
PLA2-cytosolic	Ca		PLA2-Ca*		
PIP2	PLA2-cytosolic		PIP2-PLA2*		
PIP2	PLA2-Ca*		PIP2-Ca-PLA2*		
DAG	PLA2-Ca*		DAG-Ca-PLA2*		
AA			APC		
PLA2*	Ca		PLA2*-Ca		
PLA2*			PLA2-cytosolic		
GEF*			Inact-GEF		
GTP-Ras			GDP-Ras		
GAP*			GAP		
GTP	GDP-Ras	GEF	GTP GDP-RAS GEF		
GTP Ras-GDP GEF			GDP	GTP-Ras	GEF

FIG. 13 – Tableau rassemblant l’ensemble des réactions chimiques exploitées par le modèle de Bhalla.

interactions, avec la présence des intermédiaires des réactions notamment, ce qui donne au final 80 produits dans le système, mais nous verrons plus tard comment ces données sont traitées..

## 2.2 Données pour une nouvelle implémentation informatique

### 2.2.1 Informations de Bhalla

Comme il a été dit précédemment, le modèle qui a été étudié et implémenté diffère de celui présenté de Bhalla précédemment, mais reste assez proche([1]). Pour construire le modèle sous Matlab et MAPLE, on a utilisé les réactions chimiques, voir figure 13 et enzymatiques, voir figure 14, décrites, et on a aussi identifié les molécules qui gardent une concentration constante tout au long des simulations et que l’on appellera les paramètres, voir figure 15

### 2.2.2 Informations biologiques

L’algorithme que l’on utilise marche pour le modèle sur les lipides mais, comme on l’a vu, celui-ci est de taille réduite, une dizaine de produit. La méthode utilisée pour parcourir l’ensemble des combinaisons d’éléments est la récursivité, or, d’un point de vue technique, le parcours récursif oblige à se limiter au niveau de taille de la matrice jacobienne. Plus précisément au delà de 14 produits, ce qui fait  $2^{14}$  combinaisons possibles à traiter, et donc MAPLE stoppe le programme. Il a donc fallu exploiter d’autres algorithmes, on verra comment dans la partie 3. Nous avons aussi exploité les informations de Bhalla pour répartir les éléments selon 4 modules biologiques : PLA2, PKC, Ras, voir figure 16, et MAPK, voir

Substrat	Enzyme	Produit
cRaf	PKC-act	cRaf*
GAP	PKC-act	GAP*
inact-Gef	PKC-act	GEF*
cRaf*	MAPK*	cRaf**
PLA2-cytosolic	MAPK*	PLA2*
MKP-1	MAPK*	MKP-1-ser359*
MKP-1-ser359*	MAPK*	MKP-1**
MPAK-tye	MKP-1	MAPK
MAPK*	MKP-1	MAPK-tyr
cRaf*	PP2A	cRaf
MAPKK*	PP2A	MAPKK-ser
MAPKK-ser	PP2A	MAPKK
cRaf**	PP2A	cRaf*
MAPK	MAPKK*	MAPK-tyr
MAPK-tyr	MAPKK*	MAPK*
MAPKK	cRaf*GTP-Ras	MAPKK-ser
MAPKK-ser	cRaf*GTP-Ras	MAPKK*
MAPKK	cRaf-GTP-Ras	MAPKK-ser
MAPKK-ser	cRaf-GTP-Ras	MAPKK*
MAPK-tyr	MKP-1**	MAPK
MAPK*	MKP-1**	MAPK-tyr
MKP-1-gene	nuclear-MAPK	MKP-1
APC	PLA2-Ca*	AA
APC	PIP2-PLA2*	AA
APC	PIP2-Ca-PLA2*	AA
APC	DAG-Ca-PLA2*	AA
APC	PLA2*-CA	AA
GTP-Ras	GAP	GDP-Ras

FIG. 14 – Tableau rassemblant l'ensemble des réactions enzymatiques exploitées par la modèle de Bhalla.

Paramètres
DAG
CA
PIP2
MKP-1-gene
Ubiquination
APC

FIG. 15 – Liste des paramètres du système



RAS	PKC	PLA2
GTP-Ras	PKC-cytosolic	AA
GEF*	PKC-Ca	PLA2-cytosolic
inact-GEF	PKC-Ca-DAG	PLA2-Ca*
GDP-Ras	PKC-Ca-memb*	PIP2-PLA2*
GAP*	PKC-DAG-memb*	PIP2-Ca-PLA2*
GAP	PKC-Ca-AA*	DAG-Ca-PLA2*
GTP	PKC-DAG-AA	PLA2*
GTP GDP-Ras GEF	PKC-DAG-AA*	PLA2*-Ca
GEF	PKC-basal*	PLA2-Ca*APC
GDP	PKC-AA*	PLA2-Ca*APC
PKC-act-GAP	PKC-DAG	PIP2-Ca-PLA2*APC
PKC-act-GEF	PKC-act	DAG-Ca-PLA2*APC
GAP-GTP-Ras	PKC-act-cRaf	PLA2*-Ca
	PKC-act-GAP	
	PKC-act-GEF	
	MAPK*PLA2	
	GAP-GTP-Ras	

FIG. 16 – Eléments classés comme appartenant au module biologique Ras,PKC et PLA2

cRaf*	cRaf*-GTP-Ras
cRaf-GTP-Ras	MKP-1
MKP-1**	MKP-1-ser359*
MAPK*	Nuclear-MAPK*
cRaf**	PKC-act-cRaf
MAPK*cRaf*	MAPK*PLA2
GAP-GTP-Ras	MAPK*-MKP1-ser359
MAPK-tyr	MKP-1-MAPK-tyr
MAPK	MPK-1-MAPK*
PP2A	PP2A-cRaf*
à MAPKK*	PP2A-MAPKK*
MAPKK-ser	PP2A-MAPKK-ser
MAPKK	PP2A-cRaf**
MAPKK*-MAPK	MAPKK*-MAPK-tyr
cRaf*GTP-Ras-MAPKK	cRaf*GTP-Ras-MAPKK-ser
(cRaf-GTP-Ras-MAPKK	cRaf-GTP-Ras-MAPKK-ser
MKP-1**MAPK-tyr	MKP-1**MAPK*
MAPK*MKP-1-gene	

FIG. 17 – Eléments classés comme appartenant au module biologique MAPK

figure 17.

Il est à noter que l'intersection de ces quatre modules n'est pas vide. En effet, si pour la grande majorité des éléments, il est aisé d'identifier à quel module biologique ils appartiennent, il est en revanche plus difficile de classer les complexes de molécules associant deux éléments appartenant à deux modules différents, donc en cas d'hésitation pour classer un élément, on ne fait pas de choix et on le place dans tous les modules auxquels il pourrait appartenir.

### 2.3 Intérêts d'étudier MAPK

L'un des atouts du système biologique MAPK est le nombre important de publication le concernant, ce qui permet de le considérer comme un modèle pour l'analyse de grand réseau. Partant de ce constat, la création d'outil s'appuyant dessus, sont basés sur des informations vérifiées et reverifiées.

De plus le fait que MAPK soit impliqué dans la régulation de la migration cellulaire, l'apoptose, la prolifération et la différenciation, donne à ce réseau biologique un rôle important dans la vie de la cellule, c'est pourquoi [3]. émettent l'hypothèse que si l'on pouvait modifier et contrôler cette cascade, on pourrait faire en sorte qu'en laboratoire, les cellules soient moins sensibles aux signaux d'apoptose et aussi les rendre plus sensibles aux signaux de division cellulaire, permettant ainsi de réduire les coûts de production.

Soit  $f(X) : R_+^n \rightarrow R^n$  une application différentiable de jacobien  $J$  tel que tous les mineurs principaux de  $-J$  soient positifs. L'équation  $f(X)=0$  a au plus une solution.

FIG. 18 – Théorème de Gale Nikaido

### 3 But du stage

En se basant sur les données recueillies au cours de la recherche bibliographique, il est important maintenant de connaître maintenant les informations que l'on possède sur cette quatrième méthode de détection de modules que l'on propose. Si cette méthode a déjà été exploitée avec succès sur un réseau biologique sur les lipides dans la cellule, elle n'a cependant jamais été utilisée ailleurs, et le travail informatique effectué jusqu'à présent sur cette méthode par un stagiaire en 2006, ne permet pas de travailler sur le réseau biologique MAPK. Nous allons ainsi nous intéresser aux travaux précédents sur cette méthode, avant de l'analyser

#### 3.1 Gale Nikaido

La nouvelle méthode que l'on propose pour trouver des sommets à regrouper en module est basée sur la propriété de Gale Nikaido, voir figure 18. Cette propriété s'utilise lorsque l'on a des équations différentielles. Lorsque l'on cherche à vérifier si des équations vérifient cette propriété, on veut savoir quelles sont les équations qui s'annulent en même temps. D'un point de vue biologique, cela signifie, que l'on cherche à savoir quels sont les combinaisons de sommets du graphe qui sont à l'équilibre en même temps.

Pour savoir si la matrice jacobienne ou une de ses sous matrices vérifient cette propriété, on va devoir calculer les déterminants de leurs mineurs. Un mineur, est une sous matrice particulière. En effet, pour que notre sous-matrice soit un mineur, il faut que si on sélectionne les produits  $i, j$  et  $k$  dans la matrice sur laquelle on regarde si elle vérifie Gale Nikaido, alors cela signifie que le mineur est constitué de l'intersection des lignes  $i, j$  et  $k$  avec les colonnes  $i, j$  et  $k$  et donc on aura une sous matrice de taille  $3 \times 3$ .

Une fois le mineur déterminé, il va falloir calculer la valeur de son déterminant, et si ce déterminant est négatif ou nul, alors la matrice qui contient ce mineur ne vérifie pas Gale Nikaido, mais si celui-ci est positif, on ne peut pas encore conclure tant que l'on n'a pas testé tous les autres mineurs de notre matrice.

Pour résumer en prenant un exemple simple, on part d'une matrice jacobienne de taille  $3 \times 3$   $\begin{bmatrix} A & B & C \\ D & E & F \\ G & H & J \end{bmatrix}$ , et on sélectionne une sous matrice contenant les éléments 1 et 2. On souhaite donc savoir si les équations différentielles des produits 1 et 2 admettent un unique état d'équilibre. On a donc la sous matrice :  $\begin{bmatrix} A & B \\ D & E \end{bmatrix}$  et on doit calculer tous les mineurs possibles dans cette matrice. En ayant deux éléments, le nombre de combinaisons possibles est 3 : 1, 2 ou 1 et 2.

Et donc les mineurs à tester correspondent à "A", "E" et "A.E - D.B". Si un des trois mineurs est négatif ou nul, alors la sous matrice composée des éléments 1 et 2 ne vérifie pas la propriété, mais si les 3 sont positifs, on peut alors conclure que la propriété est vérifiée.

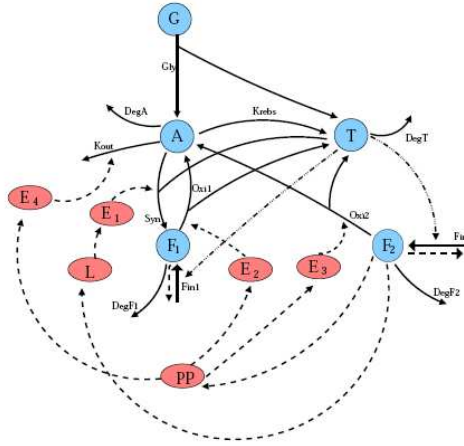


FIG. 19 – Graphe représentant le réseau lipide sur lequel la méthode basée sur la propriété de Gale Nikaido a été utilisée

### 3.2 Travail précédent

L'utilisation de cette propriété de Gale Nikaido déjà été effectuée sur un réseau lipidique par [8].

Cependant si on compare le réseau MAPK que l'on a déjà présenté au réseau sur le lipide représenté sur la figure 19, on remarque que le réseau concerné est de plus petite taille, mais les résultats obtenus sont prometteurs pour la méthode. En effet, en partant d'un réseau de 10 sommets, l'algorithme développé lors d'un stage en 2006 trouve que l'on peut regrouper tous les sommets sauf T.

En se basant sur ce résultat et en utilisant le théorème des fonctions implicites, les auteurs de la publication ont réussi à réduire le système en exprimant les équations qui s'annulent en même temps en fonction de la variable restante. Ce qui fait que d'un système à 10 équations, on arrive à une unique équation.

L'algorithme qui a été développé n'est cependant pas réutilisable en l'état, nous verrons plus tard comment on l'a modifié pour le rendre applicable au modèle MAPK. Cependant, la partie concernant la réduction ne sera pas abordée plus tard, uniquement celle concernant la recherche de plus grande boîte.

### 3.3 Différentes représentations

#### 3.3.1 MATLAB

Une première implémentation du modèle proposé par Bhalla a été fait sous MATLAB, en se basant, sur le modèle implémenté pour un autre réseau par Ovidiu Radulescu, et en entrant les informations sur les réactions connues c'est à dire la matrice stoechiométrique, ainsi que les vitesses de réactions.

Pour cela, le modèle d'implémentation proposé par Ovidiu Radulescu a nécessité une modification. En effet, à l'origine de cette implémentation, il s'agit d'un petit réseau, donc lorsqu'il faut calculer les équations différentielles (à vérifier), chacune d'elles est entrée à la main. Or avec le réseau MAPK, cela devient fastidieux. En permettant de faire ce calcul automatiquement en se basant sur la matrice stoechiométrique et sur les vitesses de réactions, on a ainsi rendu le modèle beaucoup plus souple à la réexploitation.

Bien que cette implémentation ne sera pas utilisée lors du stage, elle a offert la possibilité d'étudier la dynamique des concentrations grâce aux outils disponibles avec MATLAB. De plus, on a aussi une représentation du modèle sous la forme d'un graphe, voir figure 20, chose qui sera impossible sous MAPLE, cependant, on arrive à la même conclusion que pour la représentation visuelle sous genesis, l'ensemble reste peu compréhensible.

### 3.3.2 MAPLE

Maintenant nous allons voir dans le détail les informations qui ont été implémentées dans le modèle sous MAPK.

Ainsi en se basant sur les informations publiées par Upinder Bhalla, nous avons implémenter 80 produits différents, représentant les substrats, les enzymes, les complexes enzymes-substrats et les produits des réactions. Mais parmi ces 80, 6 sont considérés dans le modèle comme des paramètres, et donc ils ne sont pas traités de la même manière que les autres étant donné que leur concentration est fixe et donc ne sont pas intéressants pour la propriété de Gale Nikaido qui s'intéresse aux combinaisons de produits qui sont stables en même temps. Et donc au final on se retrouve avec 74 produits.

Ces produits interviennent dans 31 réactions chimiques et 28 réactions enzymatiques. Cependant, comme nous allons l'expliquer dans la partie 3.3.1, les réactions enzymatiques seront traitées en deux temps ce qui fait au total 87 réactions implémentées. On va ainsi créer une matrice stoechiométrique de taille 74 par 87 dans laquelle on indiquera par un -1 quand le produit est un réactif de la réaction et par un 1 quand le produit est un produit de la réaction.

En partant de cette matrice et des constantes de réactions, dont le détail de l'implémentation sera fait en section 3.3.1, nous avons pu créer la matrice dérivative, c'est à dire, une matrice de taille 87 par 74 où pour chaque élément de la matrice, cela correspond à la dérivée de l'équation différentielle par rapport au produit correspondant. Par exemple, si on part d'une réaction du type de la figure , on aura la matrice 1 x 3 et l'équation différentielle suivante :

$$\frac{dX_i}{dt} = K_1 \cdot X_1 \cdot X_2 - K_2 \cdot X_3$$

Ainsi, si on la dérive par rapport à  $X_1$  on obtient  $K_1 \cdot X_2$ . De la même manière en dérivant par rapport à  $X_2$  et  $X_3$ , on trouve la matrice dérivative suivante :  $\begin{bmatrix} K_1 \cdot X_2 & K_1 \cdot X_1 & -K_3 \end{bmatrix}$ . Une fois la matrice stoechiométrique et la dérivative calculées, on multiplie ces deux matrices ensemble pour obtenir la matrice jacobienne, sur laquelle on va travailler principalement, de taille 74 x 74.

### 3.4 Code général

D'un point de vue technique, la méthode que l'on va utiliser travaillera en calcul formel, ce qui fait qu'il est nécessaire de faire passer nos valeurs numériques en valeurs symboliques. De plus, l'algorithme développé pour la publication lipidique n'est pas réexploitable. Ainsi nous allons voir comment à partir d'un modèle MAPLE existant et des données numériques, l'algorithme utilisé pour ce travail a été réalisé.

p

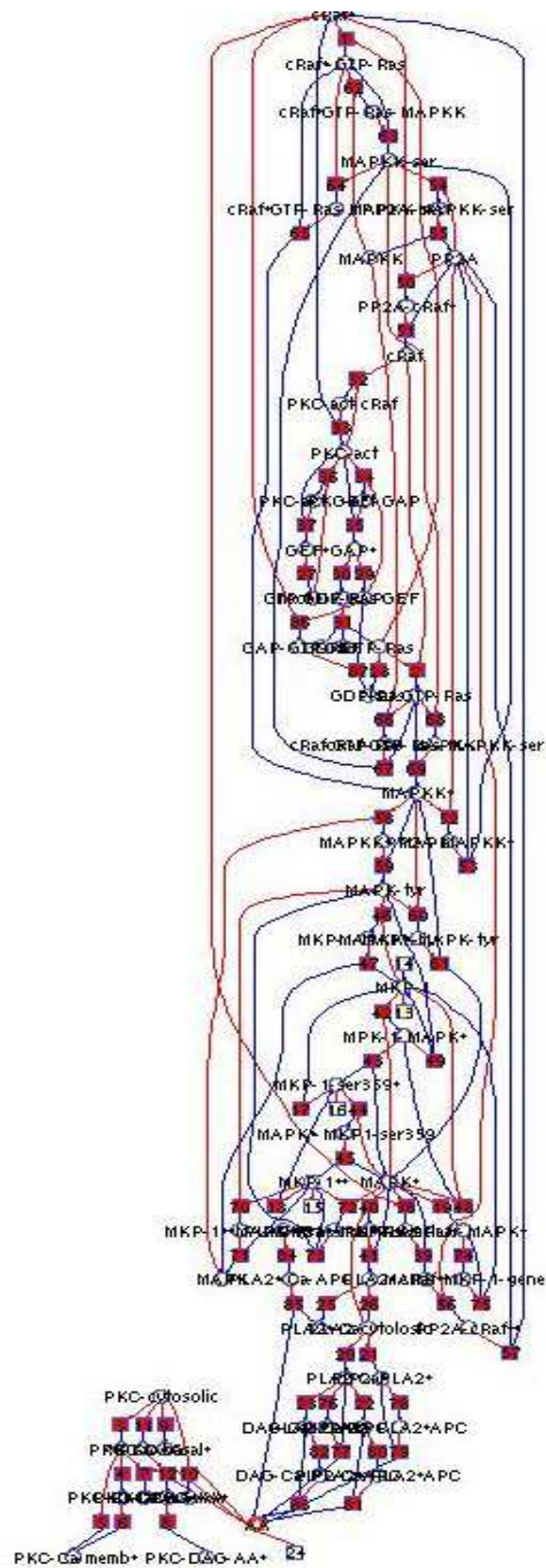


FIG. 20 – Modèle de MAPK complet.

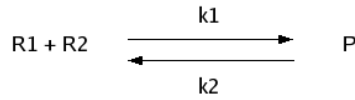


FIG. 21 – Exemple de réaction chimique avec deux réactants donnant un produit, le paramètre  $k_1$  correspond au paramètre de la réaction  $R_1 + R_2$  donne  $P$ , et  $k_2$  correspond à la réaction inverse.

### 3.4.1 Représentation symbolique

Les informations biologiques fournies par Bhalla, nous donnent plus d'informations que nous n'en utilisons réellement. En effet, nous connaissons les concentrations de départ des différents produits, et nous savons aussi lesquels garderont leur concentration constante (Buffered). Si la deuxième information nous a permis de déterminer quels étaient les éléments à classer comme paramètres, la valeur numérique ne nous sert à rien dans le cadre de l'utilisation de Gale Nikaido ?

De plus, dans le cas des réactions chimiques et enzymatiques, voir figure 21 et figure 22 en s'appuyant sur les équations de la figure 23, nous connaissons les paramètres de réactions  $k_1$  et  $k_2$  associés aux réactions chimiques et les paramètres  $k_1, k_2$  et  $k_3$  associés aux réactions enzymatiques. Or l'approche en symbolique ne nécessite pas ces informations, la méthode marche tout aussi bien avec des symboles représentant ces paramètres de réactions que les valeurs numériques. Nous adoptons donc le format suivant pour les paramètres des réactions chimiques et enzymatiques :  $K_i C_j$  pour les réactions chimiques et  $K_i E_j$  pour les réactions enzymatiques. La deuxième lettre permet bien évidemment de repérer à quel type de réaction nous avons affaire. Le  $i$  indique le paramètre de réaction, c'est à dire le sens de la réaction comme décrit sur les figures 21 et 22, avec une légère différence pour la réaction enzymatique : les réactions enzymatiques ont été séparées en deux réactions distinctes, le  $k_3$  devient donc le  $k_1$  de la deuxième réaction alors qu'on va décider que le  $k_2$  est nul. Enfin le  $j$  indique le numéro de la réaction, l'ordre des réactions est choisi de manière arbitraire, et correspond à l'ordre donné dans la publication de Bhalla. En ce qui concerne les réactions enzymatiques, comme dit précédemment, par rapport aux réactions décrites par Bhalla, on va travailler avec deux fois plus de réactions, et donc le numéro attribué à la réaction est différent de l'ordre de Bhalla : les réactions de numéros pairs correspondent donc à la deuxième partie des réactions enzymatiques.

Une autre représentation symbolique a été nécessaire pour les équations, les concentrations des produits. Dans ce cas là, le format est :  $X_n$  où  $n$  est le nom du produit. Cette représentation sert lorsque l'on calcule la matrice dérivative puis lors du calcul du jacobien. Pour ces représentations symboliques, la seule information que nous ayons est qu'ils sont, forcément, strictement positifs. Et lors que l'on effectuera nos calculs, si on lui demande le signe de  $K_1 C_1 * X_A - K_2 C_2$  le logiciel MAPLE ne pourra pas le déterminer.

### 3.4.2 Données diverses

Un des défauts principaux de la première version du code de recherche de plus grande boîte, est que le programme fonctionne mais avec du code non réutilisable. En effet, mis

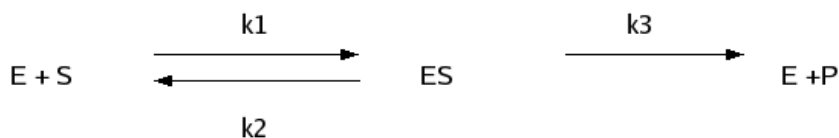


FIG. 22 – Exemple de réaction enzymatique avec une enzyme et un substrat qui s’associent en un complexe enzyme substrat, les paramètres  $k_1$  et  $k_2$ , ont la même signification que pour la réaction chimique. Le complexe enzyme-substrat donne ensuite un produit et redonne l’enzyme qui était associé dans le complexe, le paramètre  $k_3$  est le paramètre de cette réaction, et, comme décrit dans cette partie, il sera nommé  $k_1$  vu que cette réaction sera traitée comme une réaction à part.

$$\begin{aligned}
 d[ES]dt &= k_1[E][S] - k_2[ES] \\
 \frac{d[P]}{dt} &= k_3[ES]
 \end{aligned}$$

FIG. 23 – Equations des réactions enzymatiques

à part les fonctions tout le reste dont les appels de fonctions sont codés sans avoir en tête une réexploitation du code.

Une des premières étapes à consisté à proposer un nouveau code qui permette de fonctionner en ne modifiant que les données initiales, et laisser l’algorithme gérer le reste. Les informations initiales souhaitées sont donc : la matrice stoechiométrique, la liste des produits et réactants, les paramètres ainsi que d’autres valeurs numériques, comme le nombre de réactions ou le nombre d’éléments. De plus dans notre cas, nous fournissons la liste des sous modules biologiques à explorer.

A partir de ces informations, on lance l’algorithme qui va transformer ces données en matrice dérivative dans un premier temps puis en une matrice jacobienne sur laquelle on pourra faire une recherche de plus grande boîte sur les éléments qu’on aura précédemment sélectionnés.

### 3.5 Difficulté : Données plus importantes

#### 3.5.1 Différentes approches

Comme dit précédemment, l’algorithme utilisé pour traiter les données biologiques du réseau lipidique est basé sur une approche récursive, et lorsqu’il tourne sur le jeu de données contenant 10 produits, il marche de manière efficace. Cependant une fois lancé sur un nombre plus important, le temps d’exécution devient rapidement très long (188000 sec pour trouver 1 seule boîte avec une jeu de données de 13 éléments, soit 52h) et empêche d’utiliser l’ordinateur pour d’autres tâches.

Une première approche envisagée cependant, a été de modifier l’algorithme, pour permettre de faire fonctionner le code sur un plus grand jeu de données. La modification apportée a consisté à ne pas considérer toutes les combinaisons possibles à partir de  $n$  éléments dès le départ, mais de les traiter par taille décroissante. Au début, on teste donc la seule combinaison de taille max, puis les combinaisons de taille max-1 et ainsi de suite jusqu’aux combinaisons de taille 1. Si en pratique cela a permis de pouvoir lancer l’algorithme sur

Si un système contient un mineur négatif ou nul,  
alors tous sur-système ne vérifient pas la propriété de Gale Nikaido. height

FIG. 24 – Propriété permettant de réduire le nombre de calcul

une trentaine de produits, le temps de calcul devient si long, que cela rend l'utilisation de cette méthode impossible.

Il a donc fallu abandonner l'idée du parcours récursif au profit de l'utilisation des boucles pour parcourir l'ensemble des données. Une première boucle permet de passer en revue les différentes tailles de combinaisons possibles, toujours en allant de la plus grande combinaison à la plus petite. Et une deuxième boucle traite chaque combinaison possible. Ceci est rendu possible et simple d'utilisation par les fonctions MAPLE : `choose(n,j)` et `numbcomb(n,j)`, qui respectivement donnent les différentes combinaisons de  $j$  éléments parmi  $n$  et le nombre de ces combinaisons.

On va effectuer ce double parcours à deux occasions. La première lorsque l'on parcourt toutes les boîtes possibles. Pour rappel une "boîte", est une combinaison d'éléments, et cette boîte vérifie Gale Nikaido si tous ses mineurs sont positifs. Donc sur chacune des boîtes, on va tester chacun des mineurs, ce qui nous donne l'occasion de parcourir l'ensemble des combinaisons possibles à partir des données de cette boîte étant donné que chaque combinaison correspond à un mineur. De plus, par rapport à l'algorithme de départ, une amélioration apportée a été de rechercher toutes les boîtes vérifiant Gale Nikaido pour une taille donnée lorsque l'on a trouvé la première. En effet, dans l'algorithme initial, une fois que l'algorithme a trouvé une boîte, il s'interrompt, alors que dans les faits, la première boîte trouvée n'a pas plus de pertinence que celle non testée mais de taille équivalente. Si par exemple, on part d'un jeu de données de taille 3 :  $[1,2,3]$ , et que ceci ne forme pas une boîte car la combinaison  $[2,3]$  donne un mineur négatif alors que les autres sont tous positifs, il n'y a pas de raison de privilégier  $[1,2]$  plutôt que  $[1,3]$  comme plus grande boîte.

### 3.5.2 Amélioration : Gain de temps

Il a été expliqué précédemment qu'on a divisé les éléments du modèles suivant une classification parmi 4 modules biologiques, mais dans tous ces modules la taille reste encore trop importante pour faire une recherche directement de la plus grosse boîte.

Une méthode utilisée pour gagner du temps peut être décrite par une propriété, voir figure 24. Donc si on se trouve avec un système qui contient un mineur négatif ou nul, en identifiant le mineur en question, on peut ainsi écarter toutes combinaison d'éléments qui contiendrait ce mineur. Dans la pratique, on va utiliser une méthode informatique pour stocker l'information qui nous intéresse lorsque l'on rencontre un mineur négatif ou nul, c'est à dire la combinaison d'éléments ayant donné ce mineur négatif ou nul, ainsi que, mais principalement pour des raisons de codage, la taille de cette combinaison et le nombre de combinaisons ayant donné un mineur négatif jusqu'à présent. De plus, on ne va pas traiter l'ensemble du module biologique courant, mais seulement une partie. Dans les faits, en prenant l'exemple du module biologique RAS, de taille 13, un premier test a été effectué sur les 10 premiers éléments, puis un deuxième test sur les 10 derniers, puis enfin un troisième sur les 5 premiers et 5 derniers en même temps. Evidemment, l'intersection des jeux de tests n'étant pas nulle, si une combinaison appartenant à cette intersection donne un mineur négatif ou nul, on lancera le test suivant en gardant cette information, pour éviter une perte de temps.



### 3.6 Résultats

Nous allons voir les résultats obtenus par cette méthode sur le jeu de données créé en se basant sur les informations fournis par Upinder Bhalla.

Comme il a été dit précédemment, l'objectif de l'algorithme est de découvrir les variables, donc les sommets du système, qui vérifient ensemble la propriété de Gale Nikaido. En lançant le programme sur des jeux de données correspondant aux différents modules biologiques définis précédemment, on a découvert quels étaient dans chaque cas les plus grosses boîtes.

**Module Ras :** En commençant par le module biologique Ras, qui correspond à la partie du réseau qui active la cascade faisant ainsi le lien entre le module PKC et le module MAPK, ce module contenant 13 éléments nous permet de connaître quels sont les combinaisons qui donnent un mineur négatif ou nul. Ainsi, on sait en se basant sur le tableau récapitulatif pour ce module que GEF et GTP-GDP-Ras-GEF ne peuvent appartenir à la même boîte.

GEF	GTP-GDP-Ras-GEF		
GTP	GTP-GDP-Ras-GEF	GDP	
GTP	GEF	GDP	
GEF* inact-GEF	PKC-act-GEF		
GAP*	GAP	PKC-act-GAP	GAP-GTP-Ras

Comme on remarque, les combinaisons regroupent des éléments qui interagissent entre eux lors des réactions, ce qui crée un cycle, ce qui est pas compatible avec le principe de Gale Nikaido qui se base sur un état d'équilibre.

Ainsi on calcul l'ensemble des boîtes ne contenant aucune de ces combinaisons, et on trouve par exemple la combinaison suivante :

GTP-RAS
GEF*
inact-GEF
GPD-Ras
GAP*
GAP
GTP-GDP-Ras-GEF
GDP
PKC-act-GEF
GAP-GTP-Ras

Cependant il s'agit seulement d'une combinaison parmi plusieurs possibles.

A titre indicatif, si on compare le temps d'exécution du code sur la recherche de module en effectuant ou non, un calcul préliminaire sur différentes sélections du module biologique pour déterminer les combinaisons ne vérifiant pas la propriété de Gale Nikaido, on arrive en temps cumulé à 8h en ayant effectué les précalculs et à 3 jours et 10 heures sans les avoir effectués.

**Module PKC** En poursuivant la recherche de module sur le module biologique PLA2, correspondant à une partie du rétrocontrôle positif qui relie la module PLA2 au module Ras, on arrive à trouver une boîte de 16, alors que le module biologique est au départ de taille 17. En effet, la seule combinaison qui donne un mineur négatif ou nul, est celle

regroupant les sommets du réseau :

PKC-act PKC-act-cRaf PKC-act-GAP PKC-act-GEF
---

On va donc déterminer très facilement les 4 boîtes de tailles 16 qui vérifient la propriété de Gale Nikaido, il suffit de prendre tous les 13 autres sommets du réseau biologique, et 3 éléments parmi la combinaison donnant un mineur négatif.

**Module PLA2** De même en s'intéressant au module biologique PLA2, qui sert dans le rétrocontrôle positif en reliant le module MAPK et le module PKC, on arrive à trouver une boîte de taille 12 avec un module biologique au départ de taille 13. Donc de la même manière en se basant sur la combinaison :

PLA2-Ca*APC PIP2-CA-PLA2*APC DAG-Ca-PLA2*APC PLA2*-Ca
--

on détermine quels sont les 4 combinaisons qui vérifient Gale-Nikaido.

**Module MAPK** Finalement si on regarde le 4ème et dernier module biologique, qui contient 35 éléments, la détection de module est moins aisée. En effet, après avoir testé une dizaine de combinaisons de taille 10, on se retrouve tout le temps avec une aucune combinaison qui donne un mineur négatif. Au final, on n'obtient aucun moyen qui permette de réduire le temps d'exécution que l'on aura si l'on travaille directement sur les 35 éléments. Les informations recueillies laisseraient croire que ce module biologique vérifie Gale Nikaido dans son ensemble, mais du fait de sa grande taille, il est pour le moment extrêmement long de tester les  $2^{35}$  combinaisons possibles d'éléments.

## Conclusion

Le travail effectué lors de ce stage, a permis de mettre en lumière les problèmes qu'une telle méthode rencontre lors de son application à un réseau de grande taille. Si à ce jour les connaissances sur le réseau MAPK sont nombreuses d'un point de vue biologique, elles restent encore faibles d'un point de vue informatique en ne pouvant garantir que la simulation des comportements des concentrations. L'implémentation de la méthode et des données biologiques fournies par Upinder Bhalla, bien que donnant des résultats prometteurs, ne permettent pas encore de réduire le réseau à un système exploitable. Cependant, en se basant sur ce modèle et le travail déjà effectué, la possibilité de trouver de nouvelles voies pour traiter les données devraient permettre à terme de détecter les boîtes de manières plus rapide et surtout de pouvoir réduire par petites portions le réseau pour arriver à un résultat satisfaisant d'un point de vue réduction en taille.

## Références

- [1] Bhalla and Iyengar. Robustness of the bistable behavior of a biological signaling feedback loop. *Chaos*, 11(1), March 2001.
- [2] Bhalla, Ram, and Iyengar. Map kinasephosphatase as a locus of flexibility in a mitogen-activated protein kinase signaling network. *Science*, 297, August 2002.
- [3] Chapman and Asthagiri. Resistance to signal activation governs design features of the map kinase signaling module. *Wiley Periodicals*, 2004.
- [4] De Jong and Thieffry. Modélisation analyse et simulation des réseaux génétiques.
- [5] Kholodenko. Negative feedback and ultrasensitivity can bring about oscillations in the mitogen-activated protein kinase cascades. *Eur J BioChem*, 267, 2000.
- [6] De Leenheer, Angeli, and Sontag. A tutorial on monotone systems-with an application to chemical reaction networks. July 2004.
- [7] Elisabeth Pécou. Mathematical comments on basic topics in systems biology. June 2005.
- [8] Radulescu, Siegel, Pécou, and Lagarrigue. A model for regulated fatty acid metabolism in liver ; equilibria and their changes. *Springer-Verlag*, 2006.
- [9] Sivakumaran, Hariharaputran, Mishra, and Bhalla. The database of quantitative cellular signaling management and analysis of chemical kinetic models of signaling networks. *Bioinformatics*, 19(3), 2003.
- [10] Sontag. Molecular systems biology and control.
- [11] Soulé. Mathematical approaches of differentiation and gene regulation.
- [12] Yildirim and Mackey. *Biophys J.*, 84(5), May 2003.