Analyse statique et réduction de modèles pour un langage de réécriture de graphes à sites

Jérôme Feret

PSL UNIVERSITÉ PARIS

HABILITATION À DIRIGER DES RECHERCHES

DE L'UNIVERSITÉ PSL

Présentée à l'École normale supérieure

Analyse statique et réduction de modèles pour un langage de réécriture de graphes à sites

Présentation des travaux par

Jérôme FERET

Le 12 décembre 2023

Discipline

Informatique

Composition du jury :

Élisabeth, REMY Directrice de recherche,CNRS/I2M	*Présidente*
Franck, DELAPLACE Professeur, université d'Évry/Paris-Saclay	*Rapporteur*
Roberto, GIACOBAZZI Professeur, The University of Arizona	*Rapporteur*
Éric, GOUBAULT Professeur, X/Université Paris-Saclay	*Rapporteur*
Bernadette, CHARRON-BOST Directrice de recherche, CNRS/ENS/PSL	*Examinatrice*
Patrick, COUSOT Professeur, New York University	*Examinateur*
Walter, FONTANA Professeur, Harvard Medical School	*Examinateur*
Marc, POUZET Professeur, ENS/CNRS/PSL	*Examinateur*
Xavier, RIVAL Directeur de recherche, INRIA/ENS/PSL	*Examinateur*
Adelinde, UHRMACHER Professeure, Universität Rostock	*Examinatrice*

ENS | PSL

Table des Matières

Chapitre 1

Introduction

1.1 Contexte et motivations

Décrire et analyser les systèmes à grande échelle et fortement combinatoires qui sont issus de certains modèles mécanistiques de biologie des systèmes est encore hors de portée de l'état de l'art. Dans de tels modèles, le comportement individuel des occurrences de protéines, qui peuvent établir des liaisons et modifier leur capacité d'interactions, est influencé par des compétitions pour des ressources communes. De plus, les occurrences de protéines peuvent former une grande diversité de configurations d'espèces biochimiques différentes. La concurrence entre des interactions à différentes échelles de temps génère des boucles de rétro-actions non linéaires qui contrôlent l'abondance des configurations des espèces biochimiques. Enfin, ces systèmes font intervenir des interactions entre de très petites molécules, comme des ions ou des ligands et des espèces biochimiques gigantesques comme les brins d'acide désoxyribonucléique, le ribosome, ou le signalosome. Comprendre comment le comportement collectif des populations de protéines qui définit le phénotype, est engendré par le comportement individuel des occurrences de ces protéines reste un problème largement ouvert et un enjeu crucial.

Alors que les progrès technologiques permettent d'obtenir rapidement une quantité toujours plus importante de détails à propos des interactions mécanistiques potentielles entre les occurrences de protéines, et ce, à un prix très accessible, la communauté scientifique est encore bien loin de comprendre globalement comment le comportement macroscopique des systèmes émerge de ces interactions. C'est l'objectif annoncé de la biologie des systèmes. Mais ce but est sans espoir à moins que des méthodes spécifiques et innovantes pour décrire ces systèmes complexes et analyser leur propriété ne soient conçues. Bien entendu, ces méthodes devront passer à l'échelle de la très grande quantité d'informations qui est publiée dans la littérature à un rythme qui augmente de manière exponentielle.

1.2 Les langages de modélisation de systèmes d'interactions moléculaires

Les langages formels ont été beaucoup utilisés pour décrire des modèles d'interactions mécanistiques entre occurrences de protéines. Ils procurent des outils mathématiques pour traduire ces interactions et définir rigoureusement le comportement des systèmes ainsi représentés grâce à un choix de sémantiques qualitatives, stochastiques ou différentielles.

Les langages tels que les réseaux réactionnels [67] ou les réseaux de Petri classiques [95], se basent sur le paradigme de la réécriture multi-ensemble. Les interactions consistent à consommer des réactifs en échange de produits. Des constantes cinétiques permettent de préciser soit la vitesse, soit la fréquence moyenne – selon le choix de la sémantique – d'application des différentes réactions. Ceci les rend très utiles pour décrire et formaliser le comportement de systèmes d'interactions de petite ou moyenne taille. Cependant, ces langages peinent à représenter de grands modèles car ils ont besoin d'un nom (ou d'un emplacement dans le cas des réseaux de Petri) par type de configurations d'espèces biochimiques.

Des langages de plus haut niveau, inspirés des différents paradigmes de programmation, tels que les tableaux d'états à messages [49], les automates communicants [118], les algèbres de processus [122, 37], les langages orientés objet [62], les réseaux de Petri colorés [83] et la réécriture de graphes à sites [59, 65, 5, 100], exploitent le fait que les interactions dépendent généralement de conditions locales sur les configurations des occurrences de

protéines au sein des espèces biochimiques. Ces langages permettent ainsi de traduire les systèmes d'interactions entre les occurrences de protéines de manière plus parcimonieuse : seuls les détails qui importent pour une interaction donnée sont mentionnés pour décrire ces interactions.

Il est important de distinguer les approches basées sur les agents de celles basées sur les règles de réécriture. Dans celles basées sur les agents, chaque entité, que ce soit un processus [37] ou un objet [62], doit contenir la description de tous ses comportements possibles. Les changements entre les configurations des différentes entités se synchronisent par le biais de règles de communication. Ces règles, généralement en très petit nombre, définissent la sémantique opérationnelle des langages. Il est possible de conditionner le comportement d'un agent à des propriétés de l'état d'un autre agent auquel cet agent serait lié, mais cela nécessite de recourir à des processus fictifs pour aller chercher cette information. Cette astuce était en fait déjà utilisée dans les premiers modèles décrits en π-calcul [122]. Cependant, en général, les approches basées sur les agents donnent lieu à des systèmes de processus à états finis [102]. Ceci permet d'étudier leur comportement à l'aide d'outils de vérification symbolique de modèles comme PRISM [107]. Lorsque les occurrences des protéines admettent trop de configurations différentes ou lorsque leurs capacités d'interactions dépendent trop des occurrences des protéines auxquelles elles sont liées, les approches fondées sur les agents ne passent pas à l'échelle, tant au niveau de la description des modèles, que pour le calcul de leurs propriétés.

Les approches fondées sur les règles définissent les modèles par des règles d'interactions. Chaque règle définit sous quelles conditions sur les configurations des agents une interaction peut avoir lieu et quels sont les effets de cette interaction. Ainsi l'état des agents ne définit pas une fois pour toute les capacités d'interactions de cet agent. Ce sont les règles du modèle qui le font. Il n'est pas non plus nécessaire de donner la liste exhaustive de toutes les configurations des agents. Les règles peuvent se contenter de ne mentionner que les parties importantes des agents pour l'interaction qu'elles décrivent. Les approches fondées sur les règles passent mieux à l'échelle et facilitent la mise à jour des modèles. De plus, comme il n'est pas nécessaire de spécifier explicitement toutes les capacités d'interactions des occurrences des protéines, elles encouragent à une modélisation sans *a priori* où les interactions émergent des règles au fur et à mesure de la conception du modèle.

Le calcul des ambients [31, 32], des bioambients [121] et celui des membranes [30] sont un peu particuliers. Ils permettent de décrire des boîtes ou des compartiments, qui peuvent être arbitrairement imbriqués au sein d'une arborescence, alors que des agents, contenus dans les boites dans le cas des ambients, ou dans leurs parois dans le cas des membranes, permettent à ces compartiments de se déplacer ou de se fusionner. Les capacités d'interactions des agents peuvent alors dépendre de leur localisation dans la hiérarchie des compartiments. La calcul projectif des membranes [60] représente plus fidèlement la disposition des compartiments au sein d'une cellule, en rendant la description de l'état du système indépendante du choix de la racine de l'arborescence des compartiments.

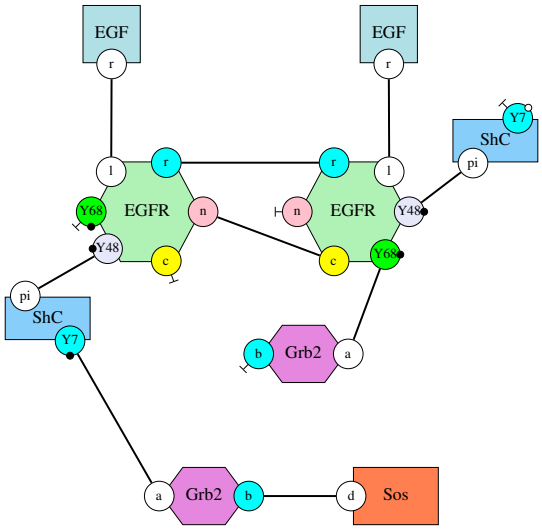
1.3 Le langage Kappa

Les langages de réécriture de graphes à sites [59, 65, 5, 100] permettent de représenter de manière transparente les réseaux d'interactions entre des occurrences de protéines grâce à leur syntaxe inspirée de la chimie.

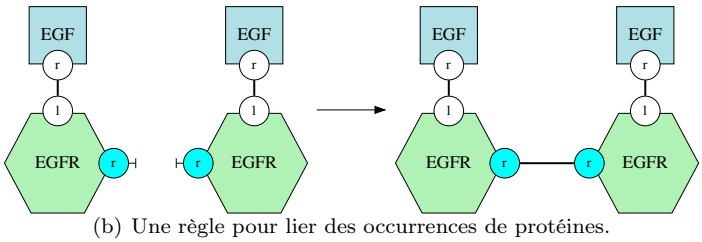
Dans Kappa, chaque configuration d'espèces biochimiques est représentée par un graphe à sites. Un exemple de graphe à sites est donné en figure 1.1(a). Dans un graphe à sites, des nœuds qui représentent des occurrences de protéines, sont associés à une liste de sites d'interactions. Ces sites peuvent être libres ou liés deux à deux. En outre, certains sites portent une propriété qui peut servir à représenter un niveau d'activation. Les interactions entre occurrences de protéines peuvent modifier leurs conformations en dépliant ou en repliant leurs chaînes de nucléotides, ce qui peut révéler ou cacher des sites d'interactions. Dans Kappa, la structure tri-dimensionnelle des occurrences de protéines n'est pas représentée explicitement. En revanche, les conditions pour qu'un site d'interactions soit visible sont spécifiées dans la description des interactions elles-mêmes.

L'évolution d'un système Kappa se décrit grâce à des règles de réécriture hors-contexte. En figure 1.1(b) est dessinée une règle pour la formation de dimères. Deux occurrences du récepteur ($EGFR$) qui sont tous deux liées à des occurrences du ligand (EGF) peuvent se lier entre elles pour former un dimer. En figure 1.1(c) est donnée une règle issue d'un modèle de réparation de l'ADN, dans laquelle l'occurrence d'une enzyme, la Glycolase (DG), peut glisser aléatoirement dans les deux sens, le long d'un brin d'ADN [104].

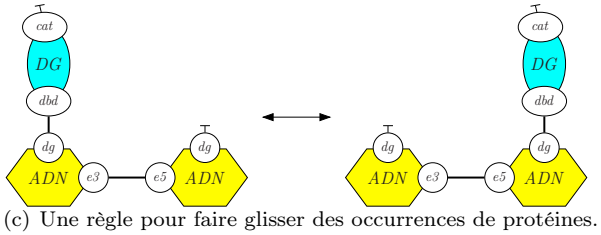
Une règle peut être comprise de manière intentionnelle comme une transformation locale de l'état du système ou de manière extensionnelle comme l'ensemble, qu'il soit fini ou non, des réactions biochimiques qui peuvent être obtenues en spécifiant entièrement les différents contextes d'application de ces règles. De cet ensemble de réactions, diverses sémantiques peuvent être définies pour décrire le comportement des systèmes. Ces séman-



(a) Un graphe à sites.



(b) Une règle pour lier des occurrences de protéines.



(c) Une règle pour faire glisser des occurrences de protéines.

Figure 1.1: En 1.1(a) est dessiné un graphe à site. Il s'agit de la configuration d'une espèce biochimique composée de deux occurrences du ligand (EGF), de deux occurrences du récepteur membranaire ($EGFR$), d'une occurrence de la protéine d'échafaudage (Shc), de deux occurrences de la protéine de transport ($Grb2$) et d'une occurrence de la protéine Sos. En 1.1(b) est donné un exemple de règle de liaison. Deux occurrences du récepteur membranaire ($EGFR$), lorsqu'elles sont toutes deux activés par une liaison avec des occurrences du ligand (EGF), peuvent se lier. Les autres sites sont omis car ils ne jouent aucun rôle dans cette interaction. En 1.1(c) est donnée une règle de déplacement. Une occurrence de l'enzyme Glycolase (DG) peut glisser dans les deux directions (selon une marche aléatoire) le long d'un brin d'ADN.

tiques peuvent être qualitatives, stochastiques ou différentielles, comme pour le cas des réseaux réactionnels et des réseaux de Pétri (les sémantiques quantitatives — stochastiques ou différentielles — requièrent l'ajout d'une constante d'interactions à chaque règle). Il est toutefois possible de simuler un modèle Kappa directement, sans passer par le réseau réactionnel sous-jacent. La simulation consiste alors à itérer la boucle événementielle suivante (celle-ci correspond à l'algorithme de Gillespie [86]). Étant donné l'état du système, représenté par un graphe à sites, l'ensemble de tous les événements possibles est calculé. Un événement consiste à appliquer une règle dans le graphe à une occurrence du motif qui constitue le membre gauche de cette règle. Chaque événement a une propension qui correspond à la constante de la règle correspondante. Le prochain événement est tiré au hasard selon une probabilité proportionnelle à sa propension, alors que le délai entre deux événements est tiré aléatoirement selon une loi exponentielle dont le paramètre est la somme des propensions de tous les événements potentiels du système. Il n'est pas raisonnable de recalculer la liste des événements potentiels à chaque fois après l'application d'une règle. Cet ensemble peut être mis à jour dynamiquement en tenant compte uniquement des nouveaux événements potentiels et des événements qui ne sont plus possibles du fait de l'application du dernier événement choisi [54]. Le simulateur actuel tire profit au maximum des sous-motifs communs dans les motifs qui apparaissent dans le membre gauche des règles pour découvrir les nouveaux événements et retirer les événements devenus obsolètes plus rapidement [17].

Le langage Kappa souffre de plusieurs limites. Par exemple, les sites d'interactions d'une même occurrence d'une protéine doivent porter des noms différents ; par ailleurs, en ce qui concerne les propriétés géométriques, Kappa ne permet ni de représenter la structure tridimensionnelle des occurrences de protéines, ni leur répartition dans l'espace. Avoir des sites deux à deux différents dans chaque occurrence de protéines facilite grandement la recherche des occurrences des motifs dans les graphes, ce qui est non seulement crucial pour simuler les modèles de manière efficace, mais est aussi à la base de plusieurs constructions utilisées pour l'analyse statique et la réduction de modèles. Certains langages lèvent cette contrainte soit directement comme dans les langages BNGL [65] et mød [4], soit indirectement en utilisant un codage sous forme d'hyperliens comme dans le langage React(C) [100]. Toutefois, l'efficacité des moteurs de simulation est fortement réduite quand de telles constructions sont utilisées. Pour ce qui est de la géométrie des protéines, les conditions liées aux conformations spatiales des protéines peuvent être encodées dans les règles de réécriture. Certaines extensions du langage permettent

de représenter des contraintes sur la position relative des occurrences de protéines et des sites d'interactions dans les configurations des espèces biochimiques, afin de restreindre l'ensemble des événements possibles à ceux qui satisfont ces contraintes [57]. Enfin, dans Kappa, la distribution des occurrences de protéines dans l'espace est passée sous silence. Il est fait l'hypothèse que les occurrences de protéines sont parfaitement mélangées. Il est donc impossible de retrouver les phénomènes d'encombrement qui peuvent être dus à des accumulations d'occurrences de protéines dans certaines régions de la cellule. De même, les gradients de concentration locaux qui pourrait être dus à la présence d'une occurrence d'une protéine d'échafaudage, ne peuvent pas être représentés (en Kappa, chaque occurrence d'une protéine d'échafaudage n'agit qu'en maintenant des occurrences de protéines dans la même espèce biochimique, une fois libérées, ces occurrences de protéines ne sont pas supposées rester, même pour un court instant dans le même voisinage). Une solution partielle consiste à encoder en Kappa une grille pour représenter de manière discrète les positions potentielles des occurrences de protéines. Ensuite, celles-ci peuvent glisser le long de cette grille grâce à des règles implémentant la diffusion des occurrences de protéines. Le langage SpatialKappa [125] utilise ce procédé de manière transparente. Par ailleurs, le langage ML [96] permet de représenter des modèles d'interactions entre occurrences de protéines qui peuvent se déplacer de manière continue dans un milieu. Il est possible de munir un modèle Kappa d'un ensemble de compartiments statiques. Toutefois, ceci ne permet pas de modéliser le transport d'occurrences de protéines par le biais de vésicules. La machine formelle cellulaire [48] répond à cet enjeu, sans toutefois fournir de moteurs de simulation efficaces.

Les langages de réécriture de graphes à sites permettent de représenter les réseaux d'interactions entre occurrences de protéines, et ce, malgré leur forte combinatoire. Si le comportement de ces réseaux peut être formellement défini et simulé, des abstractions restent nécessaires pour calculer les propriétés du comportement collectif des populations de protéines.

1.4 Interprétation abstraite

L'interprétation abstraite a été introduite il y a un peu plus de quarante ans comme un cadre mathématique pour établir des liens formels entre le comportement de programmes, vu à différents niveaux d'abstraction. Depuis, l'interprétation abstraite est utilisée non seulement pour comparer différentes méthodes et algorithmes d'analyse statique [45], mais aussi pour développer des analyseurs statiques pour calculer automatiquement les propriétés sur le comportement des programmes [8, 66]. L'interprétation abstraite s'est désormais développée dans l'industrie (entre autres, Amazon, Meta, IBM, Google, MicroSoft et MathWorks ont chacune leurs propres analyseurs statiques basés sur l'interprétation abstraite).

L'interprétation abstraite repose sur la démarche suivante. Le comportement d'un programme (ou d'un modèle) peut en général être décrit comme le plus petit point fixe $lfp\ \mathbb{F}$ d'un opérateur $\mathbb{F}$ agissant sur les éléments d'un ensemble appelé le domaine concret D. Le domaine concret est habituellement l'ensemble des parties $\wp(S)$ d'un ensemble d'éléments S, qui peuvent être des états, des traces de calcul, *et cetera*. Une abstraction est alors vue comme un changement de granularité dans la description du comportement des programmes (ou des modèles) et ce changement de granularité peut prendre en langage mathématique diverses formes telles qu'un opérateur de clôture supérieure, une famille d'idéaux, une famille de Moore ou une correspondance de Galois. Les correspondances de Galois se sont vite imposées comme l'outil le plus populaire pour décrire une interprétation abstraite. Un changement du niveau d'observation du comportement d'un programme (ou d'un modèle) peut ainsi être décrit en choisissant un ensemble $D^\sharp$ de propriétés d'intérêt. C'est le domaine abstrait. Cet ensemble est ordonné par un ordre partiel $\sqsubseteq$. Chaque élément $a^\sharp$ de ce domaine abstrait représente intentionnellement l'ensemble des éléments concrets qui satisfont cette propriété. Cet ensemble est noté $\gamma(a^\sharp)$. La fonction γ, ainsi définie, est croissante (si $a^\sharp \sqsubseteq b^\sharp$, alors $\gamma(a^\sharp) \subseteq \gamma(b^\sharp)$). Ainsi, l'ordre $\sqsubseteq$ représente le niveau d'information.

Un élément abstrait $a^\sharp$ est dit être une abstraction d'un ensemble a d'éléments concrets, si et seulement si a est un sous-ensemble de l'ensemble $\gamma(a^\sharp)$. Une correspondance de Galois est obtenue quand chaque sous-ensemble a de l'ensemble S admet une meilleure abstraction, c'est à dire, que pour chaque partie a de l'ensemble S, il existe un élément abstrait, noté $\alpha(a)$ qui est d'une part une abstraction de l'ensemble a et d'autre part, qui est plus petit (pour l'ordre $\sqsubseteq$) que n'importe quelle abstraction de l'ensemble a. Dans un tel cas, n'importe quelle fonction croissante $\mathbb{F}^\sharp$ opérant sur le domaine abstrait $D^\sharp$ et telle que $[\alpha \circ \mathbb{F} \circ \gamma](a^\sharp) \sqsubseteq \mathbb{F}^\sharp(a^\sharp)$ pour chaque élément abstrait $a^\sharp \in D^\sharp$, admet un plus petit point fixe (pour l'ordre $\sqsubseteq$) noté $lfp\ \mathbb{F}^\sharp$. De plus, la concrétisation de ce plus petit point fixe est un sur-ensemble du plus petit point fixe de la fonction $\mathbb{F}$; ainsi le comportement du programme ou du modèle peut être calculé dans le domaine abstrait au prix d'une perte potentielle d'information puisque le résultat final est un sur-ensemble de l'ensemble de tous les comportements

possibles. Par construction, l'approche est correcte : aucun comportement de la sémantique concrète n'est oublié. Par contre, quand le sur-ensemble ainsi calculé est un sur-ensemble strict, des comportements fictifs ont été introduits par l'analyse.

Le choix du domaine abstrait est crucial. Du point de vue de l'expressivité, le domaine abstrait doit permettre de décrire les propriétés d'intérêt des programmes (ou des modèles) ainsi que les propriétés intermédiaires qui sont nécessaires pour en établir la preuve de manière inductive. D'un point de vue algorithmique, ils doivent correspondre à des propriétés qui sont relativement simples à manipuler en machine. Enfin, la structure des chaînes croissantes d'éléments abstraits (pour l'ordre $\sqsubseteq$) est également importante pour que puissent être définis des opérateurs d'extrapolation précis, dans le cas où le domaine admettrait des chaînes croissantes infinies.

Plusieurs interprétations abstraites ont été proposées pour calculer automatiquement les propriétés des modèles en biologie des systèmes. Les premières ont été inspirées par les analyses de flot d'information [12, 69] et de dénombrement [113, 70] dans le π-calcul et le calcul des ambients. Ces analyses permettent de détecter avec précision dans quels compartiments des entités peuvent entrer dans des modèles-jouet de virus infectant des cellules. Elles trouvent également des exclusions mutuelles [87, 11]. Les analyses de dénombrement permettent aussi souvent de retrouver les invariants correspondant à la conservation du nombre de chaque sorte de protéines dans les réseaux réactionnels lorsque la composition des configurations d'espèces biochimiques n'est pas représentée explicitement [1, 2]. Ces invariants sont aussi appelés invariants de places dans les réseaux de Petri.

Les modèles biologiques sont fortement concurrents et souffrent de l'explosion combinatoire dans le nombre d'entrelacements potentiels des différents événements possibles. L'interprétation abstraite a été utilisée pour oublier la séquentialité dans les traces d'exécution dans les processus de frappes [115], puis plus généralement pour les réseaux asynchrones discrets booléens ou multivalués [81]. Dans les modèles réseaux booléens ou multivalués, l'interprétation abstraite a également été utilisée pour calculer une approximation des ensembles constituant des trappes [41, 103], dans lesquels les systèmes ne peuvent plus sortir une fois entrés. Ces ensembles facilitent le calcul des trajectoires périodiques des modèles. Dans les modèles de réseaux métaboliques, l'interprétation abstraite a été utilisée pour décrire une analyse de dépendances, qui calcule l'impact potentiel de l'inhibition éventuelle d'une règle sur la concentration à l'équilibre des composants du système [101, 3].

L'interprétation abstraite peut servir à la calibration d'un modèle [105], en réalisant une partition de l'espace des paramètres en trois régions : une première région dans laquelle le modèle satisfait une propriété temporelle donnée par l'utilisateur, une seconde qui ne la satisfait pas et une troisième pour laquelle l'analyse n'a pu conclure si la propriété était satisfaite ou non.

L'interprétation abstraite est également très utilisée pour le calcul des trajectoires des systèmes hybrides [88].

1.5 L'écosystème Kappa

Plusieurs outils pour analyser et manipuler des modèles Kappa sont présentés ici.

1.5.1 Analyses statiques

Un outil d'analyse statique [15], basé sur le cadre de l'interprétation abstraite, permet de calculer automatiquement certaines propriétés des modèles. Le but est d'améliorer la confiance dans les règles qui constituent le modèle. Il s'agit de retrouver des propriétés d'intérêt que le modélisateur pouvait, ou non, avoir en tête lors de la conception de son modèle ou bien de trouver des erreurs dans la modélisation.

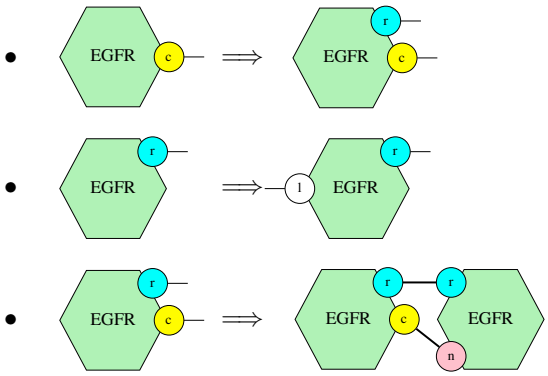
Cette analyse utilise un ensemble de motifs d'intérêt. Parmi ces motifs, l'analyse prouve que certains ne peuvent apparaître dans aucuns états potentiels du modèle. Les autres sont déclarés potentiellement accessibles : soit ils le sont effectivement, soit c'est une conséquence de la sur-approximation de l'analyse.

Les motifs d'intérêt permettent de poser des questions sur la structure biochimique des occurrences des configurations des espèces biochimiques lors de l'exécution du modèle. Actuellement, l'analyse pose trois types de questions : Existe-t-il une relation entre l'état de plusieurs sites dans les occurrences d'une protéine ? Lorsque deux occurrences de protéines sont liées entre-elles, existe-t-il une relation entre l'état de leurs sites respectifs ? Est-ce qu'une occurrence d'une protéine peut être doublement liée à une autre occurrence d'une protéine, est-ce qu'une occurrence de protéines peut être liée à des occurrences différentes d'un même type de protéines ? La première catégorie est une analyse relationnelle classique. Elle permet, par exemple, de détecter si un site ne peut être lié sans qu'un autre ne le soit ou de détecter si un site ne peut être lié sans être phosphorylé.

La seconde est utile quand des sites fictifs permettent d'encoder la localisation des occurrences de protéines, il est alors possible de vérifier, chaque fois que deux occurrences de protéines sont liées, si elles se situent nécessairement dans un même compartiment. Enfin, la troisième analyse la formation de doubles liaisons entre les occurrences de protéines. Le choix exact des questions posées par l'analyseur est fixé automatiquement suite à une inspection statique des règles du modèle.

Le résultat final de l'analyse d'accessibilité est présenté à l'utilisateur sous deux formes. D'une part, les règles dont le membre gauche est en contradiction avec les motifs qui ont été prouvés inaccessibles par l'analyse sont mentionnées à l'utilisateur. D'autre part, les propriétés intéressantes sur la structure des configurations d'espèces biochimiques sont listées sous la forme de lemmes de raffinement.

Par exemple, les trois lemmes suivants :



informent l'utilisateur que (pour le premier) dans une occurrence du récepteur membranaire, le site c ne peut être lié sans que le site r ne le soit également, que (pour le second) le site r ne peut être lié sans que le site l ne le soit aussi, et que (pour le troisième) quand une occurrence du récepteur membranaire a ses sites r et c tous deux liés, ils sont nécessairement liés tous deux à une même occurrence du récepteur membranaire.

Un lemme de raffinement est ainsi présenté comme une implication entre un motif et une liste de motifs. Ici, les listes de motifs sont toutes réduites à un élément. Il faut interpréter une telle implication par le fait que toute occurrence du membre gauche de l'implication dans un état accessible peut se raffiner dans au moins un des motifs du membre droit.

Lorsque l'utilisateur obtient des propriétés auxquelles il ne s'attend pas, il doit retourner à son modèle pour comprendre l'origine du problème. Les erreurs typographiques sont assez courantes. Il arrive aussi souvent que certaines parties du modèle manquent, il faut aller les compléter ou les remplacer par des règles fictives si l'information n'est pas disponible dans la littérature. Il se peut aussi que l'état initial du modèle ait été mal choisi. Enfin, les erreurs peuvent aussi être dues à des relations causales complexes. L'analyse statique peut alors être complétée par l'analyse causale [51, 50] et par l'analyse contre-factuelle [108] pour comprendre comment les configurations inattendues se produisent.

1.5.2 Analyses causales

La causalité est un outil très utile pour comprendre le comportement individuel des occurrences des protéines dans un modèle Kappa. Son but est d'étudier en quoi certains événements ont été nécessaires pour que d'autres événements aient pu avoir lieu.

Une trace causale est alors un ensemble d'événements dont certaines paires sont ordonnées par une relation de causalité. Celle-ci indique si l'application d'un événement a rendu possible l'application d'un autre. Un exemple de trace causale est donné en figure 1.2. Il s'agit de l'ensemble des événements pour qu'une occurrence du récepteur membranaire recrute une occurrence de la protéine *Sos* par le biais de son site *Y68*. Il faut tout d'abord activer deux occurrences du récepteur membranaire *EGFR* en les liant à des occurrences du ligand *EGF*. Les deux occurrences du récepteur peuvent alors établir une liaison symétrique, puis une liaison asymétrique ce qui permet de différencier une des deux occurrences du récepteur membranaire. Le site *Y68* de cette occurrence peut alors être phosphorylé pour qu'il puisse se lier à une occurrence de la protéine de transport *Grb2*. Indépendamment, cette occurrence de la protéine de transport peut s'être liée à une occurrence de la protéine *Sos*.

Dans cette trace, tous les événements sont nécessaires, mais d'autres *scenarii* peuvent exister. Par exemple, les occurrences du récepteur membranaire peuvent recruter une occurrence de la protéine *Sos* par le biais du site

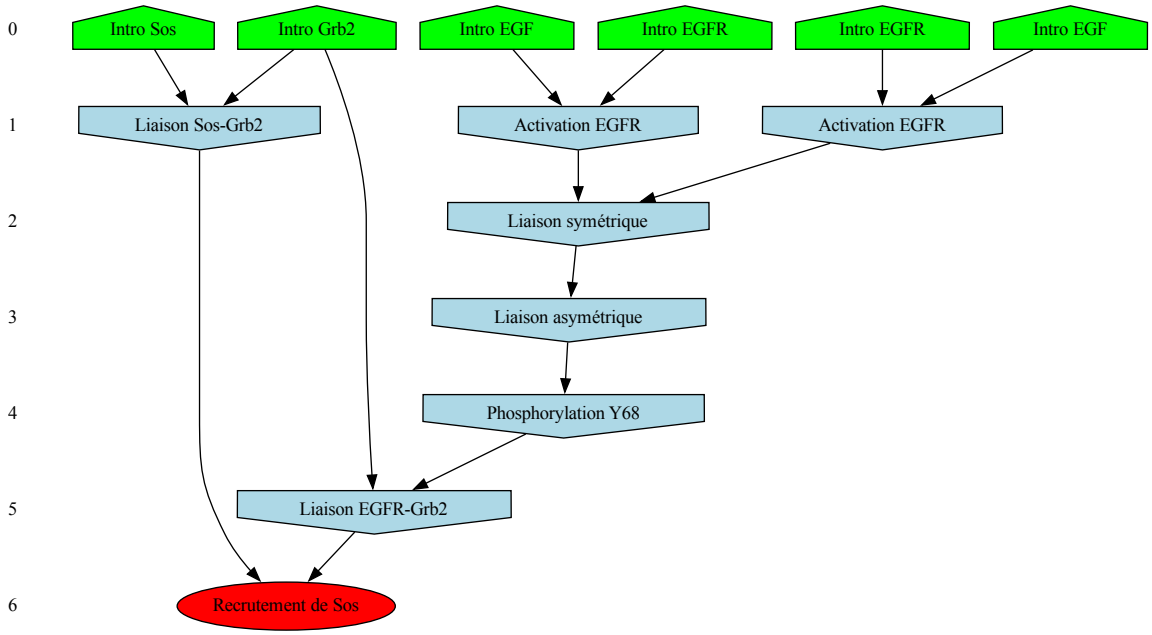


Figure 1.2: Une des deux traces causales pour le recrutement d'une occurrence de la protéine *Sos* par une occurrence du récepteur membranaire. Les nœuds verts représentent l'introduction des occurrences de protéines, les nœuds bleus représentent l'application des règles, le nœud rouge représente le but à observer. Les arcs décrivent les relations causales entre les événements.

Y48, ce qui donne lieu à une autre trace causale. Une trace causale décrit, en fait, un ensemble d'événements qui sont nécessaires dans un scénario potentiel.

Les traces causales sont obtenues en partant des résultats de la simulation, en relevant toutes les occurrences d'un événement d'intérêt. Pour chaque occurrence, une trace est extraite en collectant les événements nécessaires à cette occurrence ou récursivement à tout autre événement lui-même nécessaire [51]. Les événements sont ensuite organisés sous la forme d'un graphe acyclique orienté grâce à la transformation de Mazurkiewicz [110]. Cette transformation exploite le fait que certains événements, causalement indépendants, commutent. Un moteur de recherche opérationnelle est ensuite utilisé pour retirer de cette trace causale les événements qui peuvent l'être. Une description de cette approche dans un formalisme catégorique est décrit dans cette publication [50].

Les traces causales donnent une vision des voies de signalisation qui privilégie l'acquisition du signal. Dans un modèle, toutes les interactions sont en général réversibles, ce qui est nécessaire pour que l'occurrence d'une kinase, par exemple, puisse agir sur plusieurs occurrences de sa protéine cible à tour de rôle. Cet aspect, gestion de ressources, n'est pas du tout décrit dans les traces causales. Les traces causales ne peuvent dont pas remplacer les règles d'un modèle. Il s'agit juste d'un outil pour comprendre comment un objectif peut être atteint, mais qui ne permet pas à lui seul de définir le comportement collectif du modèle.

Les traces causales dépendent fortement de la syntaxe du langage. En effet, la syntaxe définie quelles préconditions peuvent être utilisées dans les règles, ce qui a une incidence sur le fait que deux événements puissent être vus ou non, comme indépendants causalement. Aussi, le fait que Kappa utilise de la réécriture hors contexte où seuls les sites qui ont une importance dans une interaction ont besoin d'être mentionnés, permet d'avoir plus d'événements qui commutent. Chaque trace causale peut alors résumer un plus grand nombre de traces classiques.

En figure 1.3 est considéré l'exemple d'une sorte de protéines avec deux sites de phosphorylation. Chaque site peut être phosphorylé indépendamment de l'état de l'autre site, ce qui se traduit en Kappa par les deux règles données en figure 1.3(a). Ces règles peuvent être appliquées dans n'importe quel ordre. Il y a donc une seule trace causale pour obtenir une occurrence de protéines doublement phosphorylée. Cette trace est dessinée

(a) Règles d'interactions.

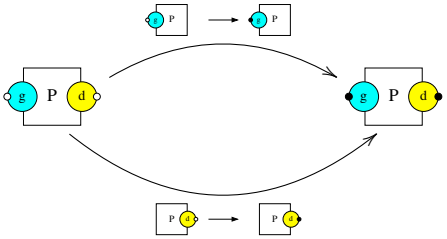
(b) Dictionnaire des configurations d'espèces biochimiques.

$$A \to C \qquad A \to B$$
$$B \to D \qquad C \to D$$

(c) Réactions.

$$A \to B \to D$$
$$A \to C \to D$$

(d) Les traces causales de ce réseau réactionnel.



(e) L'unique trace causale de cet ensemble de règle.

Figure 1.3: En 1.3(a), un modèle formé de deux règles d'interactions. En 1.3(b) l'ensemble des toutes les configurations d'espèces biochimiques accessibles à partir d'une occurrence de la protéine entièrement non phosphorylée, un nom est donné à chaque configuration. En figure 1.3(c), le réseau réactionnel sous-jacent. Contrairement aux règles d'interactions, les réactions testent l'intégralité de l'état de l'occurrence de la protéine. Ainsi, les réactions qui phosphorylent les deux sites ne commutent pas. Il y a donc deux traces causales, selon que le site droit ou gauche ait été phosphorylé en premier avec les réactions (figure 1.3(d)). En Kappa, les règles de phosphorylation d'un site s'appliquent quelque soit l'état de l'autre site. Ainsi les traces causales ne distinguent pas quel site est phosphorylé en premier. Il n'y a alors qu'un seul type de trace causale (figure 1.3(e))).

en figure 1.3(e). Dans un réseau réactionnel, les configurations d'espèces biochimiques sont nommées et leur structure biochimique ne peut pas être utilisée. Il faut donc quatre réactions pour simuler ces deux règles Kappa. Or, chacune de ces réactions spécifie exactement quel réactif elle utilise, ce qui empêche les réactions de commuter. Il y alors deux traces causales différentes selon que le site de droit ou de gauche ait été phosphorylé en premier.

Pour conclure sur la causalité, il est important de remarquer que les traces causales s'appuient sur une vision positive de la causalité. Ce n'est en général pas suffisant pour comprendre le comportement des voies de signalisation intracellulaire. En effet, il y a souvent dans ces voies des événements qui ne sont certes pas nécessaires mais qui rendent d'autres événements plus probables. C'est le cas d'une interaction qui stabiliserait une structure instable pour lui laisser le temps de réaliser une certaine interaction. D'un point de vue logique, la stabilisation de la structure n'est pas requise. Mais il est improbable que sans elle, l'autre interaction puisse avoir lieu. Ces effets cinétiques sont capturés par les notions de causalité contre-factuelles [90], dont l'adaptation à Kappa [108] ouvre des pistes de recherches pleines de promesses.

1.5.3 Réduction de modèles

La réduction de modèles consiste à simplifier un modèle en ajustant le grain d'observation. Les réductions de modèles peuvent se formaliser comme des transformations de graphes [84], des transformations tropicales [120], des bisimulations [27, 34], ou, tout simplement, des changements de variables [74]. Elles peuvent être classées selon la classe de propriétés qu'elles préservent.

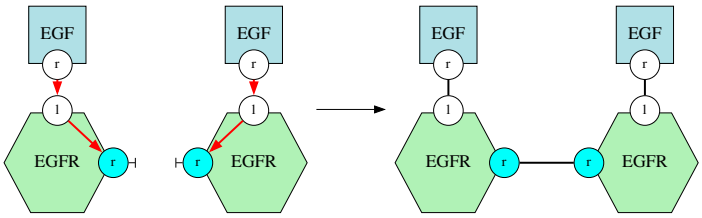
Des outils de réduction exacte permettent de simplifier à la fois les systèmes d'équations différentielles [74, 53] et les systèmes stochastiques [76] qui sont décrit en Kappa. Ces algorithmes trouvent automatiquement des changements de variables par inspection statique des règles initiales des modèles et dérivent des modèles réduits en conséquence. La preuve de correction de ces algorithmes est faite par interprétation abstraite : le modèle réduit définit la projection exacte, par le changement de variables découvert par l'analyse, du comportement transitoire du modèle avant réduction. L'ensemble des configurations des espèces biochimiques, le changement de variables et la description extensionnelle du modèle avant réduction ne sont jamais représentés explicitement, ce qui permet à la méthode de passer à l'échelle.

Les outils de réduction de modèles pour Kappa combinent deux types d'abstraction : le premièr exploite les symétries potentielles au sein des sites d'interactions des occurrences des protéines du modèle, alors que le second identifie parmi les corrélations éventuelles entre les états des sites des occurrences des protéines, celles qui n'ont aucun impact sur leur comportement collectif. Les symétries sont décrites comme des actions de groupes qui préservent l'ensemble des règles de réécriture qui constituent un modèle [27, 73]. Elles induisent une relation d'équivalence entre les configurations d'espèces biochimiques qui, elle-même, défini une relation de bisimulation sur les différents états du modèle. Les états en relation seront regroupés en un seul dans le modèle réduit. Intuitivement, cette analyse détecte quels sites ont exactement les mêmes capacités d'interactions et ignore la différence entre ces sites : dans le modèle réduit, la configuration d'une occurrence d'une protéine est définie par le nombre de sites dans un certain état en faisant abstraction de quels sites précis sont dans cet état. Ceci engendre une réduction d'un facteur exponentiel : par exemple, pour un type de protéines avec n sites symétriques pouvant chacun prendre deux états différents, la réduction permet de passer de 2^n configurations potentielles à seulement $(n+1)$.

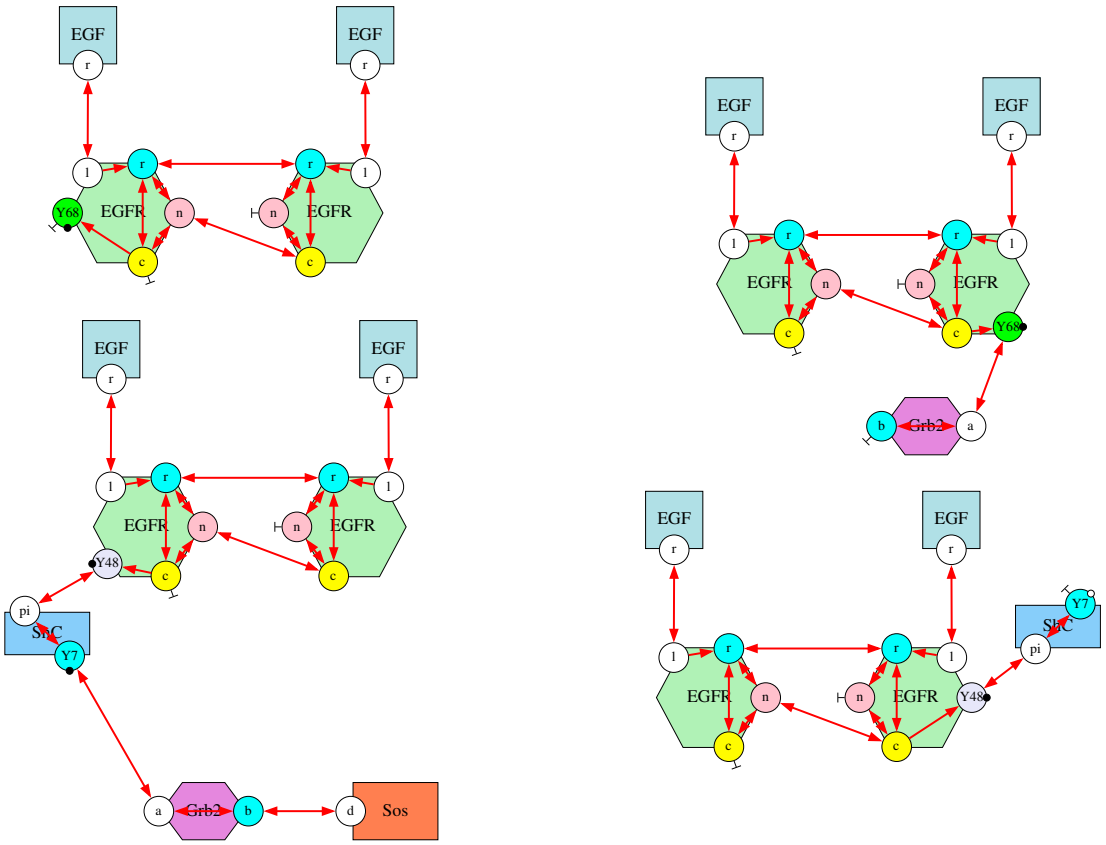
La deuxième approche se base sur l'analyse du flot d'information entre les différents sites d'interactions dans les configurations d'espèces biochimiques. Cela permet de comprendre quelles corrélations entre l'état des différents sites peuvent avoir une influence sur le comportement global du système et de passer les autres sous silence. Une approximation qualitative du flot d'information est calculée en répertoriant, au sein des règles de réécriture, tous les chemins entre les sites dont l'état est testé (ceux qui apparaissent dans le membre gauche d'une règle) et les sites dont l'état est modifié (ceux qui apparaissent dans le membre droit de cette règle avec un état différent de celui du membre gauche) (voir en figure 1.4(a)). Chaque motif est alors annoté en regroupant le flot d'information présent dans chacune des règles qui peut s'y appliquer. Les motifs intéressants sont ceux pour lesquels il existe un site d'interactions qui est accessible par tous les autres en suivant cette annotation. Par exemple, la configuration d'espèces biochimiques dessinée en figure 1.1(a) contient les quatre motifs d'intérêt donnés en figure 1.4(b) avec leur annotation. Dans ce modèle, les motifs d'intérêt sont exactement ceux qui décrivent l'état d'un seul site *Y48* ou *Y68*. Ainsi la corrélation entre l'état des différents sites *Y48* et *Y68* n'est plus représentée dans le modèle réduit. D'un point de vue combinatoire, ceci permet de passer de $m^2 \cdot n^2$ configurations d'espèces biochimiques à $m+n$ motifs d'intérêt (où m et n représentent respectivement le nombre de configurations différentes pour la partie de l'espèce biochimique liée aux sites *Y48* et *Y68*).

Sur un modèle plus complet [10, 124, 20, 51], cet outil permet de passer de 10^{20} configurations d'espèces biochimiques à $175,000$ motifs d'intérêt, en environ de 3 minutes.

Des méthodes approchées utilisent des formes tronquées de développement formels de la sémantique stochastique [85], alors que les méthodes de tropicalisation exploitent la séparation entre les échelles de temps et de concentration [119]. Ces méthodes ne procurent pas de bornes d'erreur explicites. Par ailleurs, elles nécessitent une description extensionnelle des réseaux réactionnels sous-jacents. Des méthodes exactes opèrent de manière analytique pour extraire des relations d'équivalence entre les différentes espèces biochimiques de la description explicite des réseaux réactionnels [34] ou même directement des systèmes d'équations différentielles [36]. Elles permettent de calculer la meilleure bisimulation en avant, parmi celles qui sont basées sur un partitionnement des variables ou quelles variables prennent toujours la même valeur. La notion de symétries développée pour Kappa est plus restrictive car elle se concentre sur les bisimulations qui correspondent à un certain groupe de transformations. En revanche, elle permet de détecter des relations de proportionnalité entre variables. Par ailleurs, elle ne nécessite de représenter, ni les réseaux réactionnels, ni les systèmes différentiels sous-jacents, évitant ainsi un calcul dont la durée est souvent prohibitive [29]. La réduction de modèles basée sur l'étude du flot d'information est à la fois une généralisation et une formalisation d'approches systématiques existantes [13, 40]. L'utilisation d'un langage formel et l'interprétation abstraite de sa sémantique a permis d'établir formellement la correction de ces approches.



(a) La règle de formation de dimères annotée par le flot d'information qu'elle induit.



(b) Les quatre motifs d'intérêt qui apparaissent dans les configurations d'espèces biochimiques de la figure 1.1(a).

Figure 1.4: En 1.4(a), chaque chemin entre un site dont l'état est testé et un site dont l'état est modifié dans une composante connexe du membre gauche d'une règle induit un flot d'information. Ici, la capacité de lier le site r d'une occurrence du récepteur dépend du fait que cette occurrence soit liée à une occurrence du ligand. En 1.4(b) sont représentés les quatre motifs d'intérêt qui apparaissent dans les configurations d'espèces biochimiques dessinées en figure 1.1(a). Ils sont tous quatre annotés par une relation qui spécifie comment l'information se propage – ou s'est propagée – à travers leurs différents sites d'interactions (cette relation est obtenue en recopiant le flot d'information des règles compatibles avec ces motifs). Ils contiennent chacun un site accessible par tous les autres en suivant cette relation.

1.6 Contenu

Le reste de ce mémoire décrit le langage Kappa [58, 59] sous forme graphique, ainsi que l'analyse statique qui permet de détecter quels motifs peuvent se former lors de l'exécution des modèles [56, 71, 79, 15] et des méthodes de réduction de modèles pour diminuer la complexité combinatoire de la sémantique différentielle et de le sémantique stochastique de ces modèles [74, 53, 24, 27, 26, 23, 76, 116].

En particulier, la notion de graphe à sites, qui représente l'état des systèmes modélisés, est introduite en chapitre 2, ainsi que celle de règle de réécriture. Par soucis de simplicité, seul un fragment du langage est considéré. En effet, certaines constructions du langage complet font intervenir des effets de bord (qui peuvent provoquer des transformations de l'état des occurrences de protéines, en dehors des occurrences des motifs de réécriture). S'il est possible d'adapter les différentes définitions pour traiter les effets de bords, cela n'apporte pas grand chose conceptuellement.

L'analyse statique, présentée en chapitre 3, permet de détecter, au sein d'un ensemble de motifs d'intérêt paramètre de l'analyse, lesquels ne peuvent jamais se former quelle que soit l'exécution du système. C'est une analyse approchée. Les motifs déclarés inaccessibles sont bien inaccessibles. Par contre, l'analyse n'apporte aucune information à propos des autres motifs. Par soucis d'efficacité, les ensembles de motifs sont organisés sous la forme d'une collection d'arbres de décision dans lesquels des motifs initiaux sont raffinés peu à peu en ajoutant de l'information contextuelle [79]. Cette analyse est implantée dans l'analyseur statique KaSa [15] et le choix des arbres de décisions, qui paramétrise l'analyse, est fait automatiquement par une pré-analyse.

En chapitre 4 est présentée une méthode de réduction de modèles pour la sémantique différentielle. Les sémantiques différentielles classiques introduisent une variable par catégorie d'espèces biochimiques, ce qui ne passe en général pas à l'échelle. Il s'agit donc de trouver un changement de variables qui réduise la taille de la sémantique différentielle des modèles décrit sous formes d'ensembles de règles de réécriture de graphes à sites. Pour cela, la méthode s'appuie sur la notion de flot d'information entre les sites des configurations d'espèces biochimiques, qui approche supérieurement quels sites influencent l'évolution de quels sites. Du flot d'information peuvent être découvertes des paires de sites pour lesquels la corrélation entre l'état n'a pas d'incidence sur la dynamique du système. Ces corrélation peuvent donc être oubliées, ce qui revient à considérer des morceaux de configurations d'espèces biochimiques, plutôt que des configurations d'espèces biochimiques en entier. Par construction, le résultat est un ensemble de portions de configuration d'espèces biochimiques dont l'évolution peut s'exprimer de manière autonome. L'interprétation de ces portions de configurations comme la combinaison linéaire des configurations d'espèces biochimiques dans lesquelles elles apparaissent pondérés par le nombre de leurs occurrences dans ces configurations, définit alors un changement de variables. Le système différentiel réduit, qui décrit l'évolution de la valeur de ces combinaisons linéaires peut alors être dérivé directement, sans avoir à générer le système différentiel initial.

En chapitre 5, cette analyse est portée à la réduction de la sémantique stochastique. Trois cas d'études sont donnés pour expliquer les principals enjeux de la réduction de la sémantique stochastique des modèles de réécriture de graphes à sites. Il en ressort, que la sémantique stochastique est beaucoup plus difficile à reduire que la différentielle. En effet, en stochastique, des pas de calculs peuvent agir de manière conjointe sur deux morceaux de configurations d'espèces biochimiques, ce qui impose d'en connaître les corrélations. Par ailleurs, des termes de différents ordres de grandeur apparaissent. Certains sont à l'origine de petits flots d'information qui établissent des correlations entre les états de différents morceaux de configurations des espèces biochimiques. Ces termes disparaissent dans la sémantique différentielle à la limite thermodynamique. Une analyse statique est alors proposée pour identifier des morceaux de configurations de protéines dont les occurrences se comportent de manière indépendante. Il en résulte une relation de bisimulation avant-arrière : les états équivalents à échange de morceaux d'occurrences de protéines près se comportent de la même manière vis à vis des classes d'équivalence d'états, et par ailleurs, les probabilités des états équivalents sont liées, au cours du temps, par des relations de proportionalité.

En chapitre 6 est proposé un cadre pour inférer, raisonner et exploiter les symétries potentielles dans les ensembles de règles Kappa. La présentation se concentre sur le cas des paires de sites qui exercent exactement les mêmes capacités d'interaction. Il en découle une notion de symétrie qui peut s'appliquer aux motifs, aux plongements, aux règles et aux pas de calculs, et qui définie une relation de bisimulation avant-arrière à la fois pour la sémantique différentielle et la sémantique stochastique. Ainsi, dans un modèle généré par un ensemble de règles symétriques, deux états symétriques – en sémantique différentielle – et deux distributions symétriques d'états – en sémantique stochastique – se comportent de la même manière par rapport respectivement aux classes de symétries des états et à celles des distributions d'états. Par ailleurs, si l'état initial ou si la distribution initiale des états sont symétriques, ils le restent tout au long de l'exécution du système.

Ce document se conclut en chapitre 7 et quelques perspectives sont données. La description du langage et de l'analyse reste volontairement assez haut niveau. Une formalisation complète et rigoureuse est disponible dans les articles scientifiques qui sont cités dans le corps du texte.

Chapitre 2

Réécriture de graphes à sites

Le chapitre présent décrit, dans un premier temps, la notion de graphe à sites, qui permettra de représenter à la fois les différents états possibles pour les systèmes modélisés, mais aussi, les motifs qui seront utilisés, dans un second temps, pour décrire, grâce à des règles de réécriture, l'évolution de l'état de ces systèmes.

2.1 Signature

Il faut tout d'abord définir la signature des modèles. La signature d'un modèle décrit tous les ingrédients qui peuvent intervenir dans celui-ci. Elle peut être représentée graphiquement par une *carte de contacts*, comme celle dessinée en figure 2.1. Une carte de contacts comprend des nœuds pour représenter les différentes *sortes de protéines*. Ces nœuds sont nommés et adoptent des formes et des couleurs variées pour les distinguer plus facilement. Chaque sorte de protéines est associée à un ensemble de *sites d'interaction*. Ces sites sont représentés en périphérie de chaque sorte de protéines par des cercles colorés et nommés, eux-aussi. En Kappa, une sorte de protéines donnée ne peut avoir deux sites portant le même nom. Chaque site d'interactions est associé à un ensemble de pastilles colorées qui peuvent servir à représenter son *état d'activation*, comme par exemple le fait d'être – ou non – phosphorylé ou comme le fait d'être méthylé – ou non. Un état d'activation peut aussi éventuellement servir à représenter la localisation d'une occurrence d'une protéine au sein d'un ensemble fini et fixe de compartiments cellulaires. Les sites d'interactions peuvent également porter un *état de liaison* : les sites qui portent le symbole ⊣ peuvent potentiellement rester libre ; la carte de contacts contient aussi des arcs non-orientés entre les sites qui peuvent potentiellement être liés deux à deux. En particulier, un site peut être lié à plusieurs sites dans la carte de contacts (il sera expliqué plus tard que de telles liaisons sont en compétition). Par ailleurs, un site peut être lié à lui-même dans une carte de contacts (il sera expliqué plus tard que ceci signifie que deux sites de deux occurrences différentes d'une même sorte de protéines peuvent être liés entre-eux).

Exemple 2.1.1 *En figure 2.1 est donné un exemple de carte de contacts qui correspond aux premières interactions qui interviennent dans l'activation du facteur de croissance de l'épiderme. Cet exemple est inspiré d'un modèle BNGL disponible dans la littérature [9]. Ce modèle a été étendu pour décrire la liaison asymétrique entre les récepteurs EGFR et traduit en Kappa. Cette carte introduit cinq sortes de protéines : des ligands*

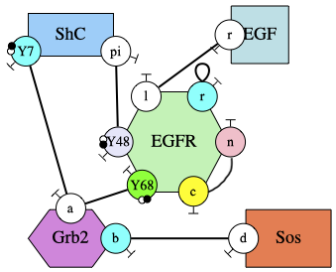


Figure 2.1: Une carte de contacts. Elle définit la signature d'un modèle en donnant la liste de toutes les sortes de protéines, leurs différents sites d'interactions, les différents états internes que peuvent prendre ces sites et les différentes liaisons potentielles entre ces sites.

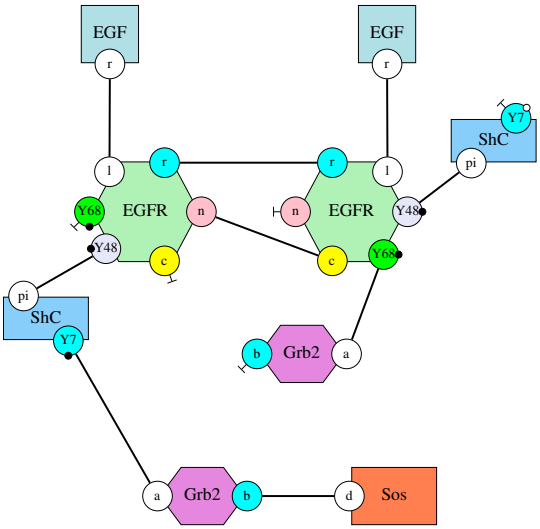


Figure 2.2: Une configuration d'une espèce biochimique. Elle contient plusieurs occurrences de protéines. Chaque occurrence documente l'ensemble de ses sites d'interactions. Les sites qui peuvent porter un état interne en ont un. Par ailleurs, les sites sont soit libres, soit liés deux à deux.

EGF, des récepteurs membranaires EGFR, des protéines d'échafaudage ShC, des protéines de transport Grb2 et des protéines cibles Sos (ces dernières seront ensuite phosphorylées ce qui initiera les étapes suivantes de la cascade d'interactions). Chaque occurrence du ligand EGF a un seul site qui est nommé r ; chaque occurrence du récepteur membranaire EGFR à six sites qui sont nommés respectivement l, r, c, n, Y48 et Y68 ; chaque occurrence de la protéine d'échafaudage ShC dispose de deux sites qui sont nommés respectivement Y7 et pi ; chaque occurrence de la protéine de transport Grb2 a deux sites qui sont respectivement nommés a et b ; enfin chaque occurrence de la protéine cible Sos a un seul site qui est nommé d. Seuls les sites Y48 et Y68 des occurrences de la protéine EGFR et le site Y7 des occurrences de la protéine ShC portent un état interne. Ces sites sont annotés par deux pastilles colorées, une blanche et une noire. La pastille blanche indique que ces sites peuvent être dans l'état non phosphorylé, alors que la noire indique que ces sites peuvent être dans l'état phosphorylé. De plus, chaque site peut être libre (symbole ⊣) ou lié. Les liaisons possibles entre sites sont entre le site r d'une occurrence de la protéine EGF et le site l d'une occurrence de la protéine EGFR ; entre les sites r de deux occurrences différentes de la protéine EGFR ; entre le site c et le n des occurrences de la protéine EGFR (il sera bientôt expliqué que la carte de contacts ne précise pas si ce doit être entre deux occurrences différentes de la protéine EGFR) ; entre le site Y48 d'une occurrence de la protéine EGFR et le site pi d'une occurrence de la protéine ShC ; entre le site a d'une occurrence de la protéine Grb2 et le site Y68 d'une occurrence de la protéine EGFR ; entre le site a d'une occurrence de la protéine Grb2 et le site Y7 d'une occurrence de la protéine ShC (il y a donc conflit entre ces deux liaisons potentielles) ; enfin entre le site b d'une occurrence de la protéine Grb2 et le site d d'une occurrence de la protéine Sos.

2.2 Configurations d'espèces biochimiques

Les modèles Kappa décrivent l'évolution d'une soupe d'occurrences de *configurations d'espèces biochimiques*. Une configuration d'espèces biochimiques est formée de plusieurs occurrences de protéines. Chaque occurrence d'une protéine est associée à un ensemble de sites d'interactions. Chaque site peut éventuellement porter un état d'activation, mais un seul. De ce fait, si un site peut être activé de deux manières différentes, avec un état de phosphorylation et un état de méthylation par exemple, ou si un site peut être doublement activé, doublement phosphorylé par exemple, il est important de définir une pastille différente pour toutes les combinaisons potentielles d'états de ce site. Enfin, chaque site doit être soit libre, soit lié à exactement un autre site. Contrairement à la carte de contacts, un site ne peut pas être lié à lui-même dans une configuration d'espèces biochimiques. De plus, un site ne peut pas être lié simultanément à deux sites. Une configuration d'espèces biochimiques forme un graphe connexe, ce qui signifie qu'il est possible de passer de n'importe quelle occurrence de protéines à n'importe quelle autre, en suivant zéro, un ou plusieurs liens.

Exemple 2.2.1 *En figure 2.2 est donné l'exemple d'une configuration d'espèces biochimiques. Celle-ci est formée de deux occurrences du ligand EGF, de deux occurrences du récepteur membranaire EGFR, de deux occurrences de la protéine d'échafaudage ShC, de deux occurrences de la protéine de transport Grb2 et d'une occurrence de la protéine Sos. Chaque occurrence du récepteur membranaire est liée au site r d'une occurrence du ligand par son site l. Les occurrences du récepteur membranaire forment un dimère grâce à une double liaison, une liaison symétrique par leurs sites r respectifs et une liaison asymétrique entre le site c de l'un et le site n de l'autre. L'occurrence du récepteur membranaire dont le site c est lié, a son site Y68 phosphorylé et libre, alors que son site Y48 est phosphorylé et lié au site pi d'une occurrence de la protéine d'échafaudage. Le site Y7 de cette occurrence de la protéine d'échafaudage est phosphorylé et lié au site a d'une occurrence de la protéine de transport dont le site b est lié au site d d'une occurrence de la protéine Sos. L'autre occurrence du récepteur a son site Y48 phosphorylé et lié au site pi de l'autre occurrence de la protéine d'échafaudage. Le site Y7 de cette occurrence de la protéine d'échafaudage n'est ni phosphorylé, ni lié à un autre site. Enfin, le site Y68 de cette seconde occurrence du récepteur membranaire est lié au site a de l'autre occurrence de la protéine de transport. Celle-ci a son site b libre.*

La signature d'un modèle restreint l'ensemble des configurations des espèces biochimiques de ce modèle. Toutes les configurations des espèces biochimiques qui sont correctes du point de vue de la syntaxe ne sont ainsi pas adéquates. Ce rôle est assuré par la carte de contacts, qui d'une part, donne la liste de tous les sites d'interactions de chaque sorte de protéines en indiquant lesquels peuvent porter un état de liaison et un état d'activation et d'autre part, résume l'ensemble des états potentiels de ces sites. Plus précisément, toute occurrence de protéines dans la configuration d'une espèce biochimique doit mentionner les mêmes sites que le nœud correspondant dans la carte de contacts. De plus, un site dont le site correspondant dans la carte de contacts admet au moins un état d'activation doit nécessairement avoir un état d'activation. Il en est de même pour l'état de liaison. Ces contraintes assurent que l'état de chaque occurrence de protéines d'une configuration d'espèces biochimiques est entièrement défini. Trois contraintes supplémentaires assurent que l'état des sites est conforme à la carte de contacts : premièrement, un site ne peut porter un état d'activation que si le site correspondant dans la carte de contacts porte également cet état d'activation ; deuxièmement, un site ne peut être libre que si le site correspondant dans la carte de contacts peut l'être lui-aussi ; troisièmement, deux sites ne peuvent être liés que si les deux sites correspondants le sont également dans la carte de contacts. Ces trois dernières contraintes peuvent se formaliser par le fait que chaque configuration d'une espèce biochimique se projette sur la carte de contacts : ainsi la fonction qui associe à chaque nœud d'une configuration d'espèces biochimiques l'unique nœud de la même sorte dans la carte de contacts doit être un *homomorphisme*. En d'autres termes, la carte de contacts peut être vue comme un repliage de toutes les configurations d'espèces biochimiques du modèle et chaque nœud de la carte de contacts résume toutes les configurations possibles des protéines du type correspondant.

Exemple 2.2.2 *En figure 2.3 est représentée la projection entre la configuration de l'espèce biochimique dessinée dans la figure 2.2 et la carte de contacts donnée en figure 2.1. Cette projection montre que cette configuration d'espèces biochimiques est compatible avec cette carte de contacts.*

2.3 Motifs

L'évolution des configurations d'espèces biochimiques est décrite par des règles de réécriture. Celles-ci définissent à la fois les conditions qui doivent être réalisées pour qu'une interaction puisse avoir lieu et les effets potentiels de cette interaction. Avant d'expliquer ce que sont ces règles de réécriture, il est nécessaire d'expliquer la notion de motifs. Celle-ci permet de spécifier sous quelles conditions une interaction peut avoir lieu.

Nous nous concentrons sur les motifs connexes. Des motifs plus élaborés peuvent être obtenus en juxtaposant plusieurs motifs connexes. Un motif connexe est une portion contigüe dans la configuration d'une espèce biochimique. De ce fait, il peut comporter zéro, une ou plusieurs occurrences de chaque sorte de protéines. Chaque occurrence de protéines est associée à un ensemble de sites d'interactions. Chaque site peut éventuellement porter un état d'activation. Enfin chaque site peut être libre, lié sans que le site auquel il est lié ne soit précisé ou lié exactement à un autre site (différent de lui-même donc). L'état de liaison d'un site peut également ne pas être spécifié.

Exemple 2.3.1 *En figure 2.4 est donné un exemple de motif connexe. Ce motif est formé de deux occurrences du récepteur membranaire EGFR et d'une occurrence de la protéine d'échafaudage ShC. L'occurrence de la*

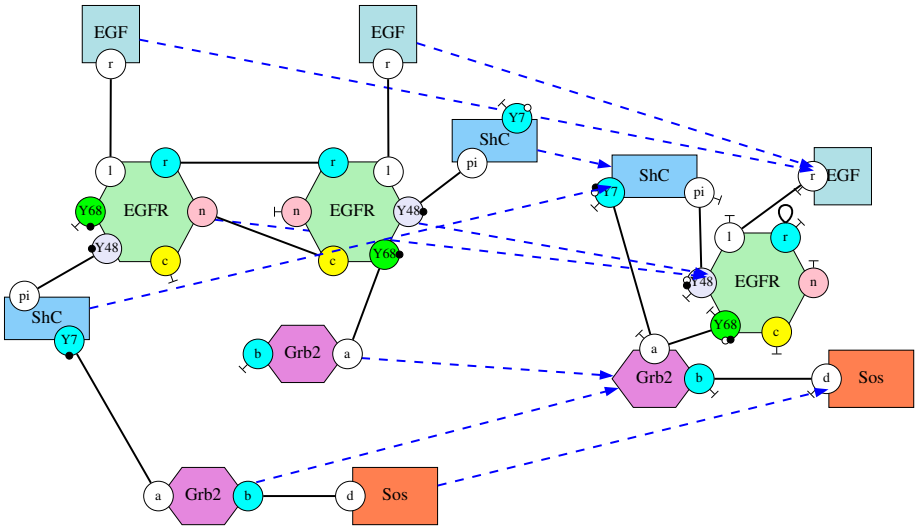


Figure 2.3: L'unique projection entre la configuration de l'espèce biochimique de la figure 2.2 et la carte de contacts de la figure 2.1. Cette projection est obtenue en associant chaque occurrence de protéines de la configuration de l'espèce biochimique à l'unique sorte de protéines correspondante dans la carte de contacts.

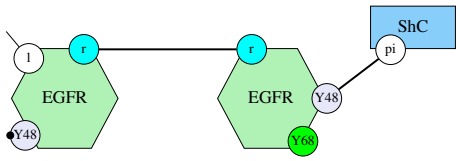


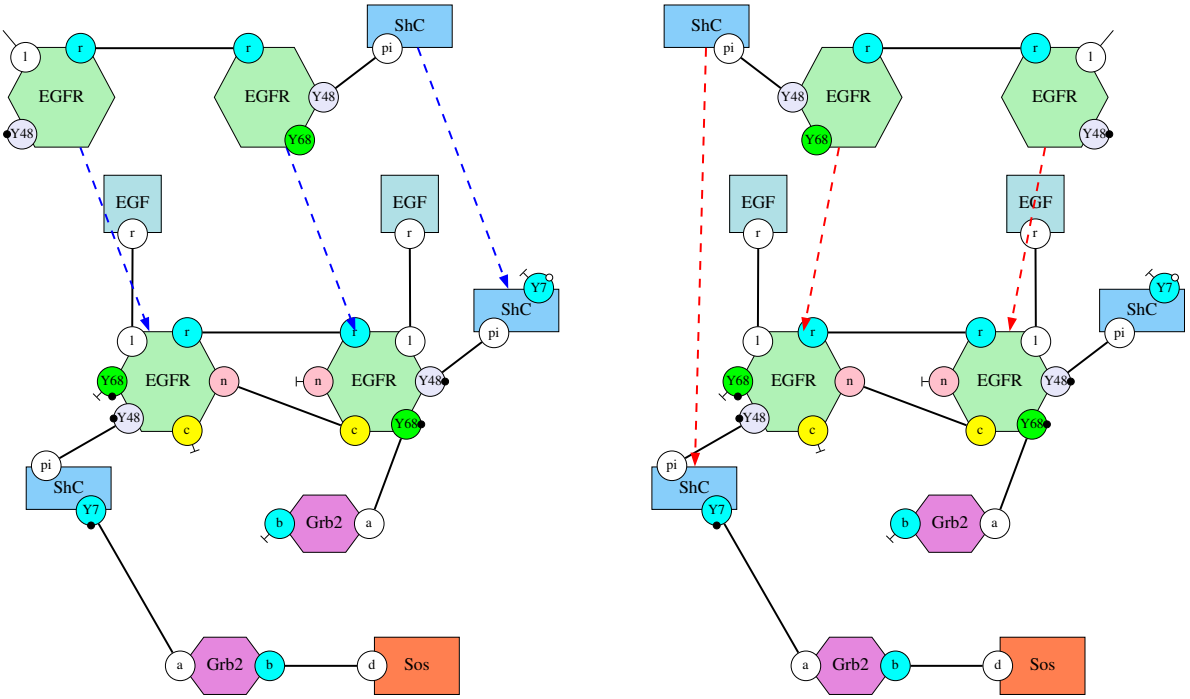
Figure 2.4: Un motif connexe. Il contient plusieurs occurrences de protéines. Chaque occurrence de protéines documente un sous ensemble de ses sites d'interactions. Chaque site peut éventuellement porter un état interne et éventuellement un état de liaison (en conformité avec la signature du modèle, donnée en figure 2.1). Comme état de liaison, un site peut être libre, lié sans que le site partenaire ne soit précisé ou être lié à un autre site.

protéine d'échafaudage mentionne uniquement son site pi. Celui-ci est lié au site Y48 d'une des deux occurrences du récepteur membranaire. L'état d'activation de ce dernier site n'est pas précisé. Cette occurrence du récepteur membranaire mentionne également son site Y68, sans en préciser ni l'état interne ni l'état de liaison, et son site r, lui-même lié au site r de l'autre occurrence du récepteur membranaire. Cet autre occurrence mentionne également son site Y48 qui est phosphorylé mais dont l'état de liaison n'est pas spécifié et son site l qui est lié à un site qui n'est pas précisé.

Comme c'était le cas pour les configurations d'espèces biochimiques, la carte de contacts contraint les motifs que l'on peut écrire dans un modèle. Ainsi, une occurrence de protéines dans un motif ne peut comporter que des sites d'interactions qui sont associés à cette sorte de protéines dans la carte de contacts. Un site ne peut porter un état d'activation que si le site correspondant dans la carte de contacts admet cet état d'activation. Un site ne peut être libre que si le site correspondant peut être libre dans la carte de contacts. Un site ne peut être lié sans préciser à quel site que si le site correspondant est lié à au moins un site dans la carte de contacts. Enfin, deux sites ne peuvent être liés ensemble que si les deux sites correspondants sont liés ensemble dans la carte de contact. En d'autres termes, comme c'était le cas pour les configurations d'espèces biochimiques, il doit être possible de projeter le motif sur la carte de contacts.

2.4 Plongements entre motifs

Un motif peut contenir plus ou moins d'information. En effet, il est possible d'ajouter des sites dans l'occurrence d'une protéine qui ne mentionne pas tous ses sites. Par ailleurs, il est possible d'ajouter un état de liaison et/ou un état interne à un site qui en manque. Il est possible de préciser à quel site un site est lié quand le partenaire de celui-ci n'est pas précisé. Il est même possible de lier un site au site d'une nouvelle occurrence de protéines.



(a) Premier plongement.

(b) Second plongement.

Figure 2.5: Deux plongements entre le motif donné dans la figure 2.4 et la configuration d'espèces biochimiques donnée dans la figure 2.2. En 2.5(a) l'occurrence de la protéine d'échafaudage est associée à l'occurrence de la protéine d'échafaudage dont le site *Y7* est libre. En 2.5(b) l'occurrence de la protéine d'échafaudage est associée à l'occurrence de la protéine d'échafaudage dont le site *Y7* est lié.

Nous dirons alors que le premier motif apparait dans le second ou encore que le second motif contient une occurrence du premier. Dans ce cas, la relation entre les occurrences de protéines du motif initial et celle du motif ainsi obtenu est formalisée par un plongement. Un plongement d'un motif vers un autre motif est une fonction qui envoie chaque occurrence de protéines du premier motif vers une occurrence de protéines du second tout en préservant la structure des graphes à sites, c'est à dire les sortes de protéines, les sites qui sont mentionnés, les états internes et les états de liaisons qui sont documentés.

Il est intéressant de remarquer que les configurations d'espèces biochimiques sont des motifs connexes particuliers. Dans ces derniers, chaque occurrence de protéines décrit tous ses sites, avec un état interne et un état de liaison quand ils en ont un. Il n'est donc pas possible d'ajouter d'information dans la configuration d'une espèce biochimique. Une configuration d'espèces biochimiques ne peut se plonger dans aucun autre motif connexe.

Exemple 2.4.1 *Deux exemples de plongements d'un motif dans une configuration d'espèces biochimiques sont donnés en figure 2.5. Ce sont les seuls plongements entre ce motif et cette configuration d'espèces biochimiques. Dans le premier (voir en figure 2.5(a)) l'unique occurrence de la protéine d'échafaudage du motif est associée à l'occurrence de la protéine d'échafaudage de la configuration d'espèces biochimiques dont le site Y7 est libre. L'occurrence du récepteur membranaire qui est liée à l'occurrence de la protéine d'échafaudage du motif est associée à l'occurrence du récepteur membranaire qui est liée à l'occurrence de la protéine d'échafaudage dont le site Y7 est libre. Enfin, l'autre occurrence du récepteur membranaire du motif est associé à l'autre occurrence du récepteur membranaire. Il est possible de remarquer que le site l de cette dernière occurrence du récepteur est lié dans le motif, sans que le site partenaire ne soit précisé, alors qu'il est explicitement lié au site r d'une occurrence du ligand dans de la configuration de l'espèce biochimique.*

Dans le second plongement (voir en figure 2.5(b)) l'occurrence de la protéine d'échafaudage du motif est associée à l'occurrence de la protéine d'échafaudage de la configuration d'espèces biochimiques dont le site Y7 est lié. L'occurrence du récepteur membranaire qui est liée à l'occurrence de la protéine d'échafaudage du motif est associée à l'occurrence du récepteur membranaire qui est liée à l'occurrence de la protéine d'échafaudage dont le site Y7 est lié. Enfin, l'autre occurrence du récepteur membranaire du motif est associée à l'autre occurrence

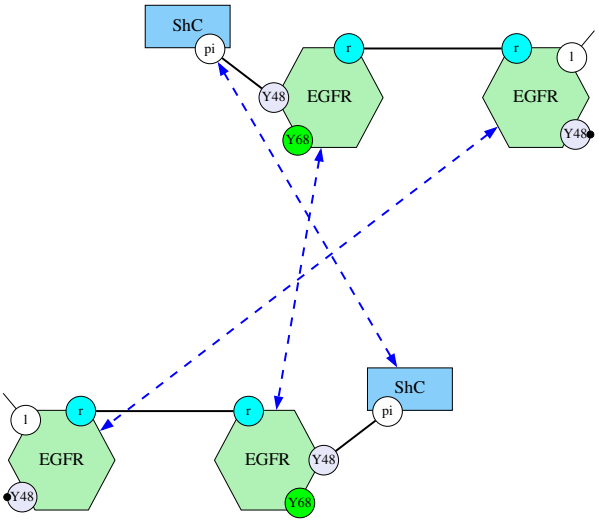


Figure 2.6: Un isomorphisme entre deux motifs. Il est possible de contruire un plongement en associant, par exemple, l'occurrence de gauche de la protéine *EGFR* du motif du bas à celle de droite du motif du haut. Réciproquement, il est possible de construire un plongement en associant, par exemple l'occurrence de gauche de la protéine *EGFR* du motif du haut à celle de droite du motif du bas.

du récepteur membranaire de la configuration de l'espèce biochimique.

Il est important de remarquer qu'un plongement d'un motif connexe vers un autre motif est entièrement caractérisé par l'image d'une occurrence de protéines. Pour avoir les autres associations, il suffit de suivre les liens et d'utiliser le fait qu'ils sont nécessairement préservés par le plongement. Cette propriété facilite la recherche d'occurrences de motifs dans les autres. Les graphes Kappa sont dits rigides [53, 117].

2.5 Isomorphismes entre motifs

Deux motifs sont *isomorphes* si et seulement si il existe un plongement de l'un vers l'autre, et réciproquement. Ces deux plongements sont alors appelés des *isomorphismes*. De ce fait, l'occurrence d'une protéine et son image par un isomorphisme contienne exactement la même information : ce sont des occurrences de la même protéine ; elles documentent les mêmes états internes et les mêmes états de liaisons ; enfin, si deux sites sont liés, leurs images par l'isomorphisme sont également liées. Un isomorphisme entre deux motifs connexes est caractérisé par l'association de l'occurrence d'une protéine du premier motif et de l'occurrence d'une protéine du second motif. Il convient alors de vérifier, en suivant les liens entre paires de sites, que cete association se prolonge bien en un isomorphisme.

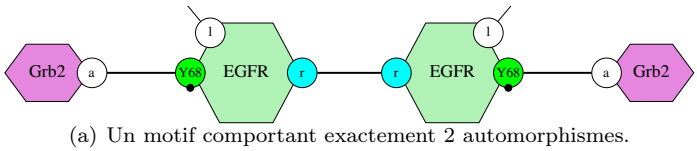
Exemple 2.5.1 *En figure 2.6 est donné un exemple d'isomorphisme entre deux motifs.*

Il s'agit d'un motif comprenant trois occurrences de protéines, deux occurrences de la protéine EGFR et une occurrence de la protéine ShC. L'occurrence de la protéine ShC document son site pi qui est lié au site Y48 de la première occurrence de la protéine EGFR. L'état interne de ce site n'est pas documenté. La première occurrence de la protéine EGFR documente également le site Y68 sans préciser ni son état de liaison, ni son état interne et le site r qui est lié au site r de la deuxième occurrence de la protéine EGFR. Cette dernière a son site l lié, sans que ne soit préciser à quel site ce site est lié. Elle documente également le site Y48 qui est phosphorylé, mais dont l'état de liaison n'est pas spécifié.

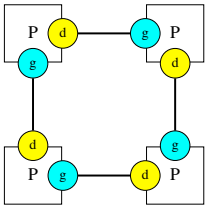
Le motif est représenté de deux manières isomorphes en inversant l'ordre dans lequel les occurrences de protéines sont dessinées. La relation entre les occurrences de protéines dans les deux représentations de motifs sont exprimées par un isomorphisme.

Par ailleurs, un plongement d'un motif dans lui-même est appelé un *automorphisme*.

Exemple 2.5.2 *En figure 2.7 sont donnés des exemples de motifs qui admettent des automorphismes non triviaux, c'est-à-dire non réduits à la fonction qui associe chaque occurrence de protéine à elle-même. Dans*



(a) Un motif comportant exactement 2 automorphismes.



(b) Un motif comportant exactement 4 automorphismes.

Figure 2.7: Deux exemples de motifs avec leur nombre d'automorphismes. En 2.7(a), le motif comporte exactement 2 automorphismes : l'automorphisme trivial et celui engendré par la transformation miroir qui échange les occurrences deux occurrences de la protéine *EGFR*. En 2.7(b), le motif comporte exactement 4 automorphismes : chaque automorphisme est engendré par l'association d'une occurrence donnée de la protéine A donnée à l'une des quatre occurrences, ce qui consiste à faire opérer une permutation circulaire sur les occurrences de la protéine A.

le premier, figure 2.7(a), un motif est composé de deux occurrences de la protéine EGFR liées entre elles par leur site r respectif. Chacune mentionne que son site l est lié (sans précisé à quel site) et que sont site Y68 est phosphorylé et lié au site a d'une occurrence de la protéine Grb2 dont le site b n'est pas documenté. Ce motif admet une symétrie mirror. En effet, il est possible d'échanger les deux moitiés du motif en échangeant les deux occurrences du site r dans les occurrences de la protéine EGFR. Dans le second, figure 2.7(b), quatre occurrences d'une protéine P sont liées circulairement par leurs sites g et d, le site g de chaque occurrence de la protéine A étant lié au site d d'une autre protéine. Il est possible d'opérer une permutation circulaire des occurrences de la protéine A tout en conservant le même motif, ce qui permet d'obtenir les 4 automorphismes du motif.

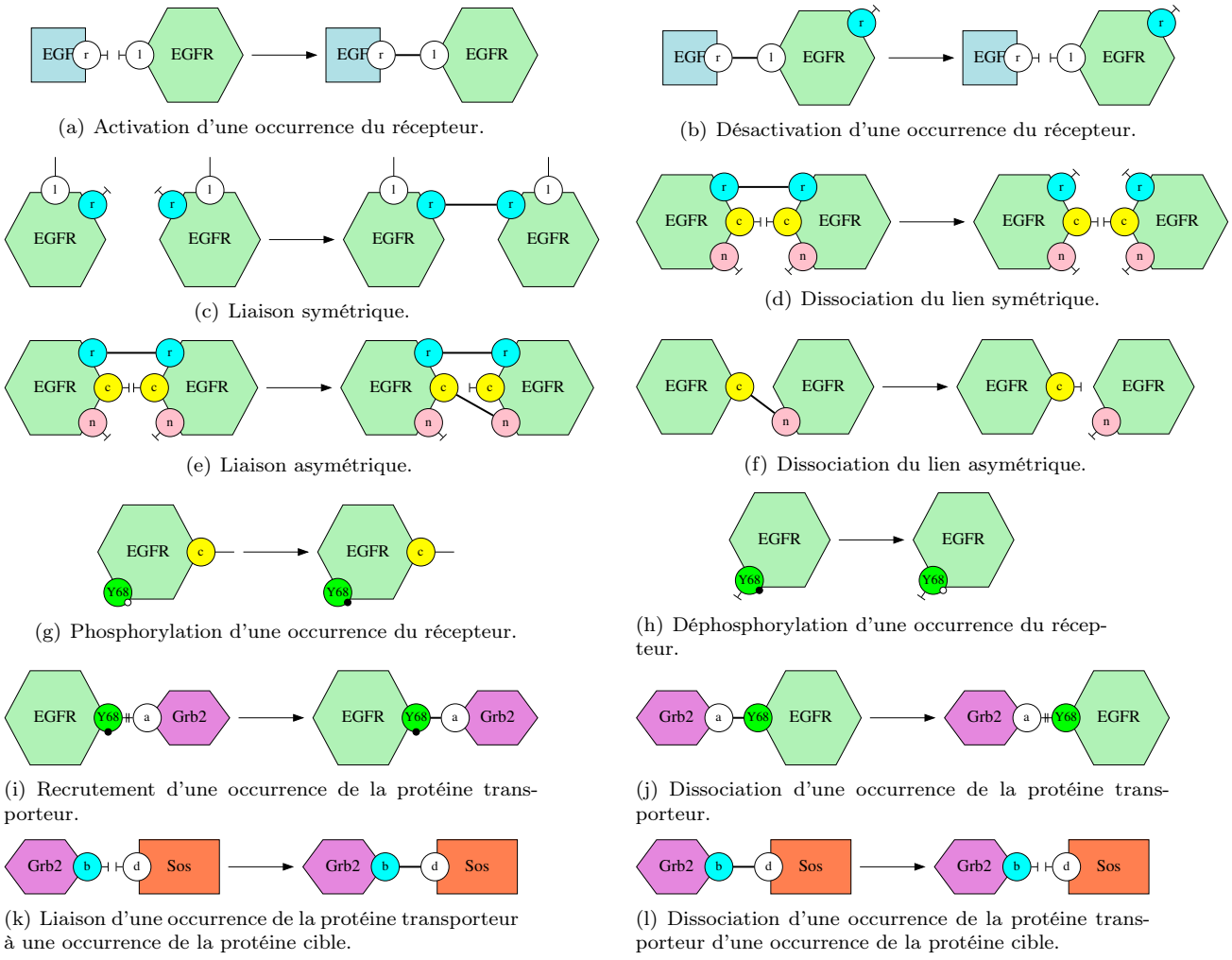
Il n'existe en fait que deux types d'automorphismes non-triviaux : les automorphismes miroir qui sont engendrés par l'échange de deux sites du même type qui sont liés entre eux (voir figure 2.7(a)) et les automorphismes circulaires qui sont engendrés par l'échange d'une partie d'un motif qui se répète en formant un anneau (voir figure 2.7(b)).

2.6 Règles d'interaction

Les motifs permettent de spécifier l'évolution potentielle de l'état des systèmes modélisés en Kappa, grâce à des règles de réécriture. Afin de simplifier la présentation, seul un fragment du langage Kappa est présenté. En particulier, les règles de réécriture qui sont introduites dans cette section n'engendrent pas d'effets de bord. Un effet de bord est une transformation à l'extérieur du membre gauche des règles. Les effets de bords peuvent être dus à des sites libérés sans préciser à quels sites ils sont liés ou à des occurrences de protéines dégradées. Ces constructions n'ont pas été considérées afin de simplifier la présentation et de présenter tous les différents concepts de la syntaxe et de la sémantique de Kappa sous forme graphique.

Les occurrences de configurations d'espèces biochimiques peuvent se transformer en appliquant des règles d'interactions. Une règle d'interaction est définie par une paire de motifs, qui contiennent exactement les mêmes sortes de protéines. Le premier motif spécifie quelles conditions locales doivent être réalisées pour permettre à l'interaction de se produire. La différence entre ces deux motifs décrit quelle transformation résulte de cette interaction. Aussi le second motif d'une règle doit pouvoir être obtenu à partir du premier en changeant uniquement l'état interne et/ou de liaison de certains sites d'interactions.

Exemple 2.6.1 *Des exemples de règles d'interactions sont données en figure 2.8. Celles-ci décrivent les interactions qui sont impliquées dans le recrutement des occurrences de la protéine cible par les occurrences du récepteur membranaire par leur site Y68, dans le modèle des premières étapes de l'acquisition du facteur de croissance de l'épiderme. Le recrutement par le site Y48 implique des règles d'interactions similaires, qui ne seront donc pas détaillées. La colonne de gauche décrit les interactions qui font progresser le recrutement d'une occurrence de la protéine cible. La première étape est l'activation d'une occurrence du récepteur membranaire par une occurrence du ligand (voir en figure 2.8(a)). En se liant à une occurrence du ligand, une occurrence du récepteur change de conformation et peut alors établir une liaison symétrique avec une autre occurrence du*



(a) Activation d'une occurrence du récepteur.

(b) Désactivation d'une occurrence du récepteur.

(c) Liaison symétrique.

(d) Dissociation du lien symétrique.

(e) Liaison asymétrique.

(f) Dissociation du lien asymétrique.

(g) Phosphorylation d'une occurrence du récepteur.

(h) Déphosphorylation d'une occurrence du récepteur.

(i) Recrutement d'une occurrence de la protéine transporteur.

(j) Dissociation d'une occurrence de la protéine transporteur.

(k) Liaison d'une occurrence de la protéine transporteur à une occurrence de la protéine cible.

(l) Dissociation d'une occurrence de la protéine transporteur d'une occurrence de la protéine cible.

Figure 2.8: Règles d'interactions impliquées dans le recrutement d'une occurrence de la protéine cible par la voie de signalisation courte (sans passer par la protéine d'échafaudage).

récepteur qui doit pour cela être elle-même activée (voir en figure 2.8(c)). Comme seules les occurrences du ligand peuvent se lier aux sites l des occurrences du récepteur, il n'est pas nécessaire de mentionner les occurrences du ligand dans la règle. Il suffit d'écrire que les sites l des deux occurrences du récepteur doivent être liés sans préciser à quels sites. Après cette étape, les deux occurrences du récepteur tiennent le même rôle. Pour les distinguer, une liaison asymétrique peut alors s'établir (voir en figure 2.8(e)) entre le site c d'une des deux occurrences et le site n de l'autre occurrence. Le site Y68 de l'occurrence du récepteur qui est liée par son site c peut alors se faire phosphoryler par l'autre occurrence du récepteur membranaire (voir en figure 2.8(g)). Cela change la conformation de cette occurrence du récepteur membranaire et lui permet de se lier à une occurrence de la protéine de transport (voir en figure 2.8(i)). Indépendamment, les occurrences de la protéine de transport peuvent se lier aux occurrences de la protéine cible (voir en figure 2.8(k)).

Chacune de ces interactions est réversible. Cependant les interactions inverses ne peuvent s'effectuer que sous certaines conditions. Ces interactions sont décrites dans la colonne de droite. Les liaisons symétriques entre les occurrences du récepteur membranaire capturent les occurrences du ligand qui ne peuvent alors pas se libérer (voir en figure 2.8(b)). Les liaisons asymétriques empêchent les liaisons symétriques de se briser (voir en figure 2.8(d)). Les liaisons asymétriques peuvent se briser sans condition (voir en figure 2.8(f)). La phosphorylation du site *Y68* d'une occurrence du récepteur est bloquée quand ce site est lié (voir en figure 2.8(h)). Les liaisons entre les occurrences du récepteur et les occurrences de la protéine de transport d'une part, et celles entre les occurrences de la protéine de transport et celles de la protéine cible d'autre part, peuvent se défaire sans condition (voir en figures 2.8(j) et 2.8(l)).

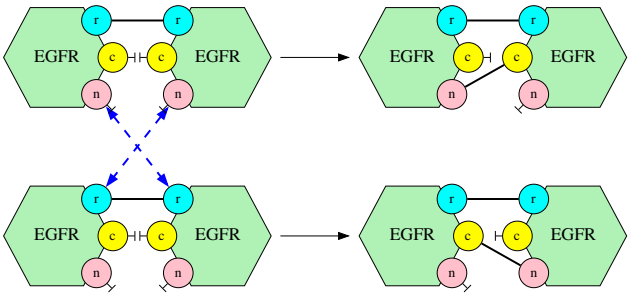
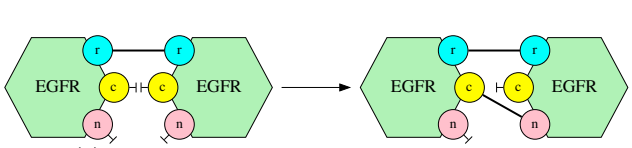
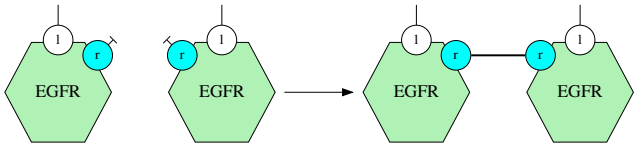


Figure 2.9: Un exemple d'isomorphisme entre deux règles. Il s'agit de la règle qui vient ajouter une liaison asymétrique entre le site c et le site n respectif de deux occurrences de la protéine *EGFR* déjà liées par leurs sites r. Cette règle d'interaction existe sous deux formes isomorphes selon que l'occurrence de la protéine *EGFR* qui lie son site n soit à droite ou à gauche du membre gauche de la règle.



(a) Une règle qui n'admet qu'un automorphisme.



(b) Une règle qui admet exactement 2 automorphismes.

Figure 2.10: Deux exemples de règles avec leur nombre d'automorphismes. En 2.10(a), la règle d'interaction ne comporte que l'automorphisme trivial. En 2.10(b), la règle d'interaction comporte exactement 2 automorphismes : l'automorphisme trivial d'une part et l'automorphisme qui consiste à échanger les deux occurrences de la protéine *EGFR* d'autre part.

Dans le langage complet, il est possible de détruire un lien entre deux occurrences de protéines en ne spécifiant qu'un seul des deux sites de liaisons. De plus, une règle peut également détruire des occurrences de protéines. Ces constructions peuvent induire des effets de bord, puisqu'appliquer de telles interactions est susceptible de libérer des sites qui ne sont pas décrits dans les membres gauches des règles correspondantes. Par ailleurs, le langage complet permet aussi de synthétiser de nouvelles occurrences de protéines.

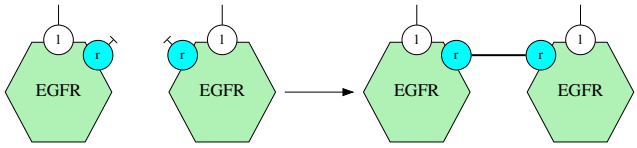
2.7 Isomorphismes entre règles

Il est possible de changer les membres gauches et droits d'une règle d'interaction par un isomorphisme commun. La règle obtenue et la règle initiale seront alors dites *isomorphes*.

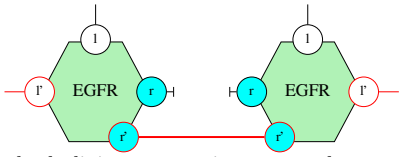
Exemple 2.7.1 *En Fig. 2.9 est donné un exemple d'isomorphisme entre deux règles d'interaction. La règle en question est la règle de liaison asymétrique (voir en figure 2.8(e)) qui ajoute une liaison asymétrique entre le site c et la site n de deux occurrences de la protéine EGFR à la condition que celles-ci soient déjà liées par leurs sites r. Dans la règle initiale, l'occurrence de la protéine EGFR qui lie son site c est à gauche dans le membre gauche de la règle. En échangeant les deux occurrences de la protéine EGFR dans le membre gauche et dans le membre droit de la règle, une autre règle est obtenue, isomorphe à la première.*

Comme pour les motifs, un isomorphisme d'une règle vers elle-même est appelé un *automorphisme*.

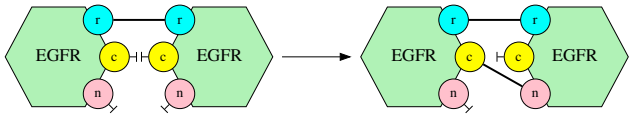
Exemple 2.7.2 *Deux exemples de règles sont donnés en figure 2.10, avec leurs nombres d'automorphismes. La première règle d'interaction, en 2.10(a) est celle de la liaison asymétrique entre deux occurrences de la protéine EGFR. Ces deux occurrences ne sont pas dans le même état dans le membre droit de la règle. En effet, la première a son site c lié, alors qu'il est libre dans la seconde. La règle ne comporte donc qu'un automorphisme : l'automorphisme triviale qui ne procède à aucun échange d'occurrences de protéines. La second règle d'interaction, en 2.10(b) est celle de la liaison symétrique entre deux occurrences de la protéine EGFR. Cette fois-ci les deux occurrences peuvent être échanger dans le membre gauche et dans le membre droit*



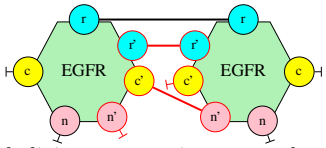
(a) La règle de liaison symétrique entre deux occurrences de la protéine A.



(b) La règle de liaison symétrique entre deux occurrences de la protéine A dessinée sous la forme d'un motif.



(c) La règle de liaison asymétrique entre deux occurrences de la protéine A.



(d) La règle de liaison asymétrique entre deux occurrences de la protéine A dessinée sous la forme d'un motif.

Figure 2.11: Règles vues comme des motifs dont l'interface des occurrences de protéines a été dupliquée. En 2.11(a) est rappelée la règle pour former une liaison symétrique entre deux occurrences de la protéine A. En 2.11(b) est représentée la même règle, mais sous forme de motif unique. Des sites ont été ajoutés à chaque occurrence du motif du membre gauche pour représenter le membre droit. Leurs noms ont été primés et leurs contours sont dessinés en rouge. Leurs états sont également dessinés en rouge. Le motif obtenu admet exactement deux automorphismes, quitte à échanger les deux occurrences de la protéine A ou non. C'est effectivement le nombre d'automorphismes admis par la règle représentée par ce motif. En 2.11(c) est rappelée la règle pour former une liaison asymétrique entre deux occurrences de la protéine A. En 2.11(d), cette règle est représentée sous forme d'un motif. Ici encore, l'état du membre droit de la règle a été ajouté au membre gauche en rouge avec des sites aux noms primés. Ce dernier motif n'admet que l'automorphisme trivial, comme c'était effectivement le cas également avec la règle correspondante.

sans changer la règle d'interaction. La règle admet donc exactement deux automorphismes : l'automorphisme trivial d'une part, et l'automorphisme qui consiste à échanger les deux occurrences de la protéine d'autre part.

Il est en fait possible de voir les isomorphismes et les automorphismes de règles comme des isomorphismes et des automorphismes de motifs, quitte à dupliquer chaque site dans la signature du modèle, avec une version du site documentant l'état du site dans le membre gauche de la règle et une autre documentant celui dans le membre droit.

Exemple 2.7.3 *En figure 2.11 sont donnés deux exemples de règles et leurs représentations sous forme de motifs uniques. Cette dernière s'obtient en dupliquant les sites qui apparaissent dans les occurrences de protéines dans les membres de la règle. Ainsi pour chaque site apparaissant dans une occurrence de protéines dans la règle représentée, deux sites figurent dans l'occurrence correspondante du motif : un site avec le même nom qui représente le site et son état dans le membre gauche de la règle et un site avec le nom primé et le contour en rouge qui représente lui le site et son état dans le membre droit. L'état des sites primés est également dessiné en rouge.*

La première règle donnée en exemple est celle de formation des liaisons symétriques entre deux occurrences de la protéine EGFR. Que ce soit dans la représentation sous forme de règle (voir en figure 2.11(a)) ou dans celle sous forme de motif unique (voir en figure 2.11(b)), les deux occurrences de la protéine peuvent être échangées sans modifier la règle ou le motif. Il y donc bien exactement deux automorphismes dans la règle et dans sa représentation sous forme de motif unique. La seconde règle est celle de la formation des liaisons asymétriques dans les configurations du dimère de la protéine A. Il n'y a qu'un automorphisme dans la règle car l'état du site n n'est pas le même dans les deux occurrences de la protéine A dans son membre droit (voir en figure)2.11(c)). De même, dans la représentation sous forme de motif unique, il n'y a qu'un automorphisme car l'état du site n' n'est pas le même dans les deux occurrences de la protéine (voir en figure 2.11(d)).

2.8 Réactions induites par une règle d'interaction

Comme signalé précédemment, le membre gauche d'une règle d'interactions spécifie dans quel contexte cette interaction peut avoir lieu. Il est alors possible d'ajouter des contraintes sur les conditions d'application d'une

règle en raffinant les motifs qui apparaissent dans les membres gauches et droits des règles exactement de la même manière. Une règle d'interactions qui ne peut plus être raffinée (sans ajouter de nouvelles composantes connexes) est alors appelée une règle-réaction [93].

Exemple 2.8.1 *En figure 2.12 est montré un exemple de deux raffinements d'une même règle d'interactions en deux règles-réactions. La règle d'interactions est celle qui permet de casser, en l'absence de lien asymétrique, le lien symétrique entre deux occurrences du récepteur membranaire (voir en figure 2.8(d)).*

1. *Dans le premier raffinement (voir en figure 2.12(a)), la règle est appliquée à un dimère dont la première occurrence du récepteur est liée par son site Y48 à une occurrence de la protéine d'échafaudage dont le site Y7 est libre et phosphorylé et par son site Y68 à une occurrence de la protéine de transport elle-même liée à une occurrence de la protéine cible. La deuxième occurence du recepteur de ce dimère est liée par son site Y48 à une occurrence de la protéine d'échafaudage dont le site Y7 est libre et non phosphorylé et par son site Y68 à une occurrence de la protéine de transport dont le site b est libre.*

2. *Dans le second (voir en figure 2.12(b)), les deux occurrences de la protéine d'échafaudage ont été interverties. Ainsi, la règle est appliquée à un dimère dont la première occurrence du récepteur est liée par son site Y48 à une occurrence de la protéine d'échafaudage dont le site Y7 est libre et non phosphorylé et par son site Y68 à une occurrence de la protéine de transport elle-même liée à une occurrence de la protéine cible. La seconde occurrence du récepteur de ce dimère est liée par son site Y48 à une occurrence de la protéine d'échafaudage dont le site Y7 est libre et phosphorylé et par son site Y68 à une occurrence de la protéine de transport dont le site b est libre.*

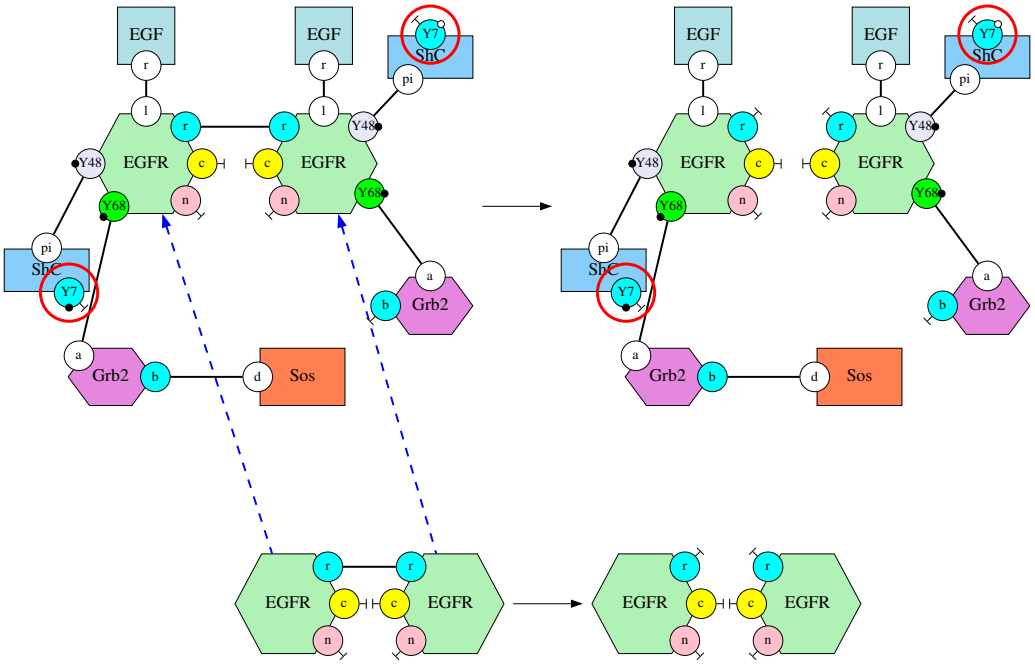
Bien que les deux configurations d'espèces biochimiques qui apparaissent dans les membres gauches de ces deux règles-réactions soient formés exactement des mêmes occurrences de protéines et dans les mêmes configurations, puisque seul l'agencement entre ces occurrences change, il apparait que les deux règles-réactions obtenues ne produisent pas les mêmes configurations d'espèces biochimiques. Ceci justifie pleinement le choix, dans Kappa, de représenter la topologie des liens entre les occurrences de protéines. Sans celle-ci il est impossible de décrire fidèlement la séparation des occurrences de récepteurs, tout en respectant la distribution des différentes occurrences de protéines et de leurs configurations dans chacune des configurations d'espèces biochimiques résultant de cette séparation. Par exemple, dans le langage BCS [61], les configurations d'espèces biochimiques sont représentées par l'ensemble des occurrences de protéines qui les constituent, ainsi que leurs configurations, mais sans préciser la topologie des liens entre ces occurrences de protéines. Aussi il est impossible de représenter fidèlement la règle de déliaison qui est dessinée en figure 2.8(d) dans ce langage.

2.9 Réseaux de réactions sous-jacents

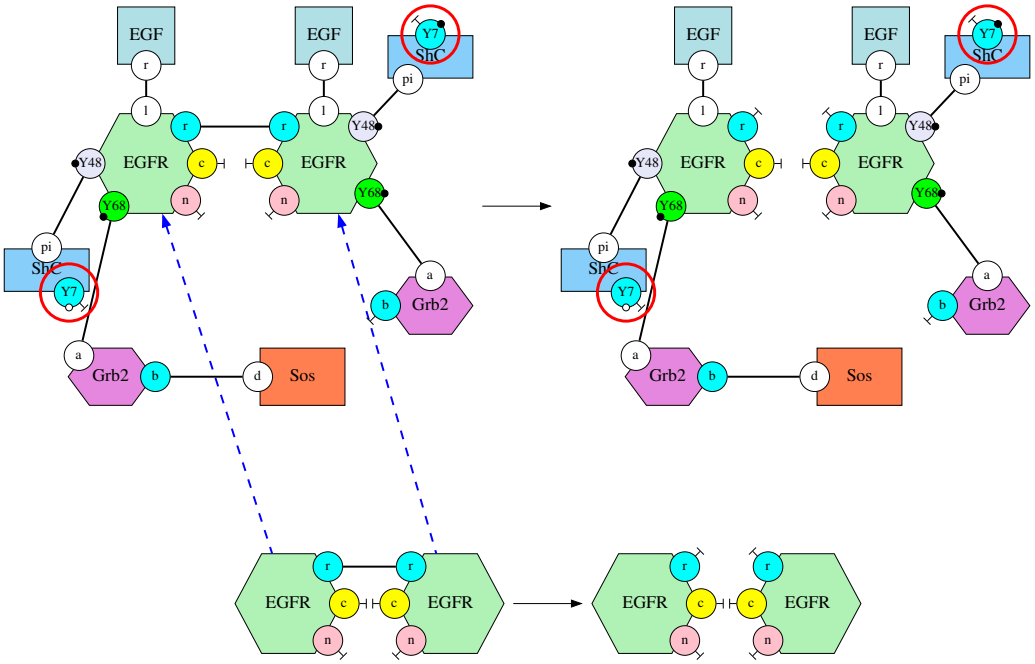
Un ensemble de règles peut alors être traduit en un ensemble – éventuellement infini – de règles-réactions en remplaçant chaque règle d'interactions par l'ensemble des règles-réactions qui peuvent être obtenues comme raffinement de ces règles. Ensuite, quitte à donner un nom à chaque configuration d'espèces biochimiques qui peut intervenir dans les règles-réactions ainsi obtenues, ces règles-réactions peuvent être assimilée à un réseau de réactions (éventuellement infini), dans lequel chaque réaction est spécifiée par une liste de réactifs et une liste de produits parmi un ensemble d'espèces biochimiques représentées uniquement par des noms (en passant sous silence leurs structures biochimiques). Ce réseau de réactions est défini de manière unique modulo le choix des noms associés aux espèces biochimiques.

Exemple 2.9.1 *Pour conclure cette section, nous détaillons la génération d'un réseau de réactions à partir d'un ensemble jouet de règles Kappa. Nous considérons un modèle avec une seule sorte de protéines qui admet deux sites, g et d, chacun pouvant être phosphorylé ou non. La signature du modèle est donnée par la carte de contacts qui est dessinée en figure 2.13(a). Le phosphorylation et la déphosphorylation de chaque site dans une occurrence de protéines peut se faire indépendamment de l'état de l'autre site, ce qui est formalisé dans les quatre règles données en figure 2.13(b). Ainsi ni les règles de phosphorylation, ni celles de déphosphorylation d'un site, ne mentionnent l'état de phosphorylation de l'autre site.*

Les règles-réactions associées à ce modèle jouet s'obtiennent en explicitant dans quel contexte local elles peuvent s'appliquer. Ici chaque règle Kappa donne lieu à deux règles-réactions selon que le site qui n'est pas mentionné dans la règle initiale est phosphorylé ou non. Ces règles-réactions sont données en figure 2.13(c).

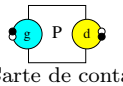


(a) Premier raffinement.



(b) Second raffinement.

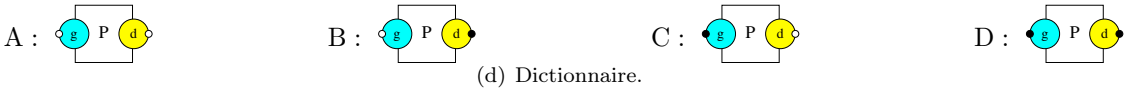
Figure 2.12: Deux exemples de raffinements d'une même règle d'interactions en deux règles-réactions. Les différences entre ces deux raffinements sont mises en valeur par des cercles rouges (les deux occurrences de la protéine *ShC* ont été échangées). Dans les deux cas, la règle-réaction est obtenue en ajoutant dans le membre gauche et dans le membre droit de la règle d'interactions exactement la même information sur le contexte d'application de la règle.



(a) Carte de contacts.

(b) Règles d'interactions.

(c) Règles-réactions.



(d) Dictionnaire.

$A \to C$
$A \to B$
$B \to D$
$C \to D$
$C \to A$
$B \to A$
$D \to B$
$D \to C$

(e) Réactions.

Figure 2.13: Un modèle formé d'une carte de contacts et de quatres règles d'interactions et sa traduction sous forme réseau réactionnel.

La prochaine étape est de nommer les différentes configurations d'espèces biochimiques qui interviennent dans les règles-réactions ainsi obtenues. La configuration de la protéine dans laquelle aucun site n'est phosphorylé, est appelée A, celle dans laquelle seul le site d est phosphorylé, est appelée B, celle dans laquelle seul le site g est phosphorylé, est appelée C et celle dans laquelle les deux sites sont phosphorylés, est appelée D. Les réactions données en figure 2.13(e) sont obtenues en remplaçant chaque occurrence de configuration d'espèces biochimiques par son nom dans les règles-réactions.

Le choix d'une sémantique en terme de réseaux réactionnels a été fait pour simplifier la présentation. C'était ainsi que le langage BNGL avait été implanté initialement [9]. Une telle sémantique est toutefois assez peu utile en pratique, car un modèle Kappa engendre en général un trop grand nombre de réactions. Par contre, la sémantique de Kappa peut être formalisée directement, soit sous forme d'une algèbre de processus [56, 76], soit dans un cadre catégorique [50, 73]. La première méthode est plus opérationnelle alors que la seconde abstrait au contraire beaucoup de détails. Il faut cependant noter que les cadres catégoriques usuels de la réécriture de graphes, que ce soit par des sommes amalgamées simples [109], par des sommes amalgamées doubles [43] ou par des sommes amalgamées et demi [42]) ne représentent pas fidèlement les effets de bord avec la définition usuelle des plongements entre graphes à sites. Deux approches connues permettent d'y remédier. Il est possible soit de changer la définition des plongements [50, 73], soit d'enrichir les objets de la catégorie par des contraintes [6].

La simulation d'un modèle Kappa opère directement par réécriture du graphe qui représente l'état du système, sans avoir à considérer le réseau de réactions sous-jacent [54, 17].

Chapitre 3

Analyse des motifs accessibles

Si la carte de contacts (e.g. voir en figure 2.1 à la page 13) donne un aperçu rapide de toutes les interactions potentielles entre les différents sites des occurrences des protéines dans un modèle, elle n'est en général pas suffisante pour décrire précisément la structure de la configuration de ses espèces biochimiques. En effet, l'état des différents sites d'interactions de la configuration d'une espèce biochimique est souvent contraint par des invariants structurels. Par exemple, dans le modèle des premières étapes de l'acquisition du facteur de croissance de l'épiderme, les sites *Y48* et *Y68* des occurrences du récepteur membranaire, ainsi que le site *Y7* des occurrences de la protéine d'échafaudage, ne peuvent être liés à un autre site sans être phosphorylés (à moins que ce soit le cas dans l'état initial). Par ailleurs, lorsque les deux sites r et c d'une occurrence du récepteur sont liés simultanément, ils sont nécessairement liés respectivement au site r et au site n d'une même occurrence du récepteur (ce qui forme une double liaison). Un autre exemple concerne les modèles avec des compartiments, comme, par exemple, une cellule dont on distingue le noyau du cytoplasme. La localisation de chaque occurrence de protéines peut alors être spécifiée comme l'état d'activation d'un site fictif. Dans de tels modèles, toutes les occurrences de protéines de la même occurrence d'une espèce biochimique sont en général localisées dans un même compartiment, ce qui se traduit par la contrainte que le site fictif de deux occurrences de protéines liées entre elles doit toujours être dans le même état. Dans certains cas, il est toutefois possible d'avoir des espèces biochimiques transmembranaires avec des portions localisées dans des compartiments voisins, c'est à dire de part et d'autre d'une membrane.

Dans ce chapitre est décrite une analyse statique qui permet de détecter automatiquement ces contraintes. Le but est de vérifier que les propriétés auxquelles peut s'attendre le modélisateur sont bien vérifiées ou bien de détecter certaines erreurs de modélisation. En particulier, cette analyse permet de trouver des *règles mortes*. Ce sont des règles qui ne peuvent jamais s'appliquer dans un modèle, car les contraintes qui sont exprimées dans leurs membres gauches ne sont pas réalisables. C'est souvent la conséquence d'erreurs typographiques (par exemple, quand une même sorte de protéines est désignée par deux noms différents dans l'encodage d'un modèle), d'un état initial incomplet, d'interactions manquantes dans le modèle (par exemple, quand l'activation d'un site n'est pas décrite, alors qu'elle est nécessaire pour la suite de la cascade d'interactions) ou de conditions causales plus complexes qu'il faut alors élucider.

Cette analyse est implantée dans l'analyseur statique KaSa [15] et intégrée dans la plate-forme de modélisation en ligne dédiée au langage Kappa [19]. Ceci permet d'assister le modélisateur pendant l'écriture du modèle en lui fournissant les contraintes structurelles qui sont vérifiées par les configurations des espèces biochimiques et en l'avertissant de la présence de règles mortes, après chaque ajout ou modification d'une règle d'interactions.

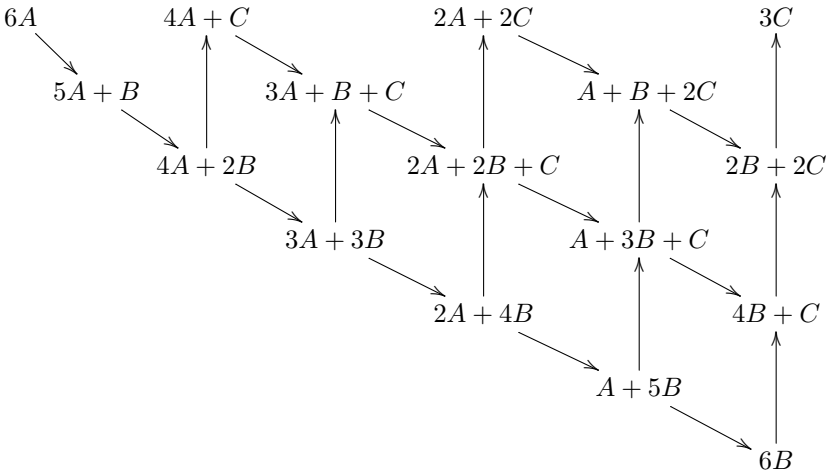
3.1 Accessibilité dans un réseau réactionnel

La première étape consiste à définir l'ensemble des états accessibles dans un modèle Kappa. Comme nous l'avons vu dans la section 2.9 page 23, un modèle Kappa induit un réseau réactionnel, ce qui permet de définir directement l'ensemble des états accessibles d'un modèle Kappa sans recourir à des constructions compliquées.

Soit un réseau réactionnel, c'est à dire un ensemble d'espèces biochimiques $\mathcal{S}$ et un ensemble de réactions $\mathcal{R}$. Chaque réaction est donnée par deux listes d'espèces biochimiques : ses réactifs et ses produits. Ce réseau induit un système de transitions dans lequel l'*état* du réseau est défini comme un certain nombre (éventuellement nul) d'occurrences de chacune des espèces biochimiques – c'est à dire une fonction de l'ensemble $\mathcal{S}$ vers l'ensemble $\mathbb{N}$

$$\begin{cases} A \to B \\ 2B \to C \end{cases}$$

(a) Un réseau réactionnel.



(b) Système de transitions sous-jacent.

Figure 3.1: Un réseau réactionnel et sa sémantique. En 3.1(a) un réseau réactionnel formé de deux réactions. La première permet de transformer une occurrence de l'espèce biochimique A en une occurrence de l'espèce biochimique B, la seconde permet de transformer deux occurrences de l'espèce biochimique B en une occurrence de l'espèce biochimique C. La restriction de l'ensemble de toutes les transitions possibles aux états qui sont atteignables à partir d'un état initial formé de six occurrences de la protéine A est dessinée en 3.1(b) sous la forme d'un système de transitions.

des entiers naturels — et les *transitions* permettent de sauter d'un état à un autre en consommant les réactifs d'une réaction et en ajoutant les produits de cette même réaction (en tenant compte de leurs multiplicités respectives). Une transition n'est possible que si l'état courant du système contient tous les réactifs qui sont nécessaires à la réaction (en tenant compte, une nouvelle fois, de leurs multiplicités respectives). Une transition d'un état q vers un autre état q' est alors notée $q \to q'$.

Exemple 3.1.1 *Un système de transitions est donné comme exemple en figure 3.1(b). Il correspond à la restriction du système de transitions associé aux réactions qui sont données en figure 3.1(a) aux états accessibles à partir d'un état initial formé de six occurrences de l'espèce biochimique A. Dans ce réseau, la somme entre le nombre d'occurrences de A, de B et deux fois celui de C est toujours égal à la quantité initiale de A. En effet, cette quantité n'est modifiée par l'application d'aucune des réactions du réseau.*

Étant donné un ensemble d'états initiaux potentiels, $\mathcal{I} \subseteq \mathcal{S}^{\mathbb{N}}$, nous définissons l'ensemble des états accessibles comme étant ceux susceptibles d'être atteints à partir d'un état initial (de l'ensemble $\mathcal{I}$) en appliquant un nombre arbitraire (éventuellement nul) de transitions. Cet ensemble peut se définir comme le plus petit point-fixe de la fonction suivante :

$$\mathbb{F} : \begin{cases} \wp(\mathcal{S}^{\mathbb{N}}) & \to \wp(\mathcal{S}^{\mathbb{N}}) \\ X & \to \mathcal{I} \cup \{ q' \mid \exists q \in X,\ q \to q' \}\,. \end{cases}$$

Il faut noter que la fonction $\mathbb{F}$ est croissante, ce qui signifie que si X_1 et X_2 sont deux ensembles d'états tels que l'ensemble X_1 soit un sous-ensemble de l'ensemble X_2, alors l'ensemble $\mathbb{F}(X_1)$ est nécessairement un sous-ensemble de l'ensemble $\mathbb{F}(X_2)$ lui-aussi. Comme, de plus, cette fonction est définie sur l'ensemble des parties d'un ensemble, le *théorème de Tarski* [127] assure que la fonction $\mathbb{F}$ admet un point fixe, plus petit que tout autre point fixe de $\mathbb{F}$. Ce plus-petit point fixe, que l'on note $\mathit{lfp}\,\mathbb{F}$, est en fait l'ensemble des états accessibles.

Malheureusement, le calcul de ce plus petit point fixe peut être coûteux, voire ne pas terminer. Ceci motive la construction d'abstractions pour calculer un sur-ensemble des états accessibles en un temps de calcul acceptable.

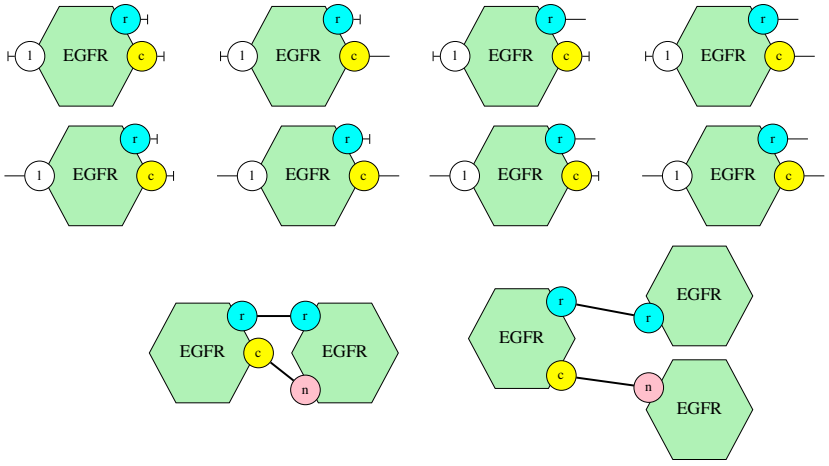


Figure 3.2: Un ensemble de motifs d'intérêt pour l'analyse des configurations accessibles des espèces biochimiques dans le modèle des premières interactions qui interviennent dans l'acquisition du facteur de croissance de l'épiderme. Les huit premiers motifs permettent de s'intéresser aux relations potentielles entre l'état de liaison des sites l, r et c dans les occurrences du récepteur membranaire. Ces 8 motifs correspondent exactement à chaque combinaison possible pour l'état de ces 3 sites, chacun de ces sites pouvant être libre ou lié. Les deux derniers motifs permettent de distinguer deux occurrences du récepteur liées par une double liaison d'une chaîne d'au moins trois occurrences du récepteur.

3.2 Abstraction d'un ensemble d'états

Lorsqu'un réseau est induit par un modèle Kappa, la structure biochimique associée aux espèces de ce réseau peut être utilisée pour construire une abstraction. Une possibilité consiste à choisir un ensemble de motifs connexes afin d'abstraire les ensembles d'états par le sous-ensemble parmi ces motifs de ceux qui apparaissent au moins une fois dans au moins un état de cet ensemble. Le choix des motifs connexes considérés est important : il définit le compromis entre l'expressivité de l'abstraction, c'est à dire son niveau d'approximation, et sa complexité, c'est à dire le coût pour effectuer des calculs à ce niveau d'abstraction.

Exemple 3.2.1 *Un exemple de motifs d'intérêt pour le modèle des premières interactions de l'acquisition du facteur de croissance de l'épiderme est donné en figure 3.2. Les huit premiers motifs se concentrent sur l'analyse des relations potentielles entre l'état des sites l, r et c dans les occurrences du récepteur membranaire. Ils correspondent à toutes les combinaisons syntaxiquement possibles pour l'état de liaison de ces* 3 *sites. Ce sont des* vues locales *(ou plus précisément des sous-vues locales) [56]. Elles permettent d'abstraire un ensemble de configurations d'espèces biochimiques par l'ensemble de toutes les configurations potentielles de toutes ses occurrences de protéines, vues indépendamment les unes des autres. Ceci revient à garder uniquement l'information à propos de l'état de liaison et l'état d'activation de chaque site dans chaque occurrence de protéines tout en passant sous silence à quel site chaque site lié l'est.*

La formation de dimères dans ce modèle fait intervenir des doubles liaisons. Il est légitime de se demander s'il est possible de former des chaines comportant successivement au moins trois occurrences du récepteur membranaire. C'est le but des deux derniers motifs de l'ensemble. Ils permettent de distinguer le cas d'une double liaison entre deux occurrences du récepteur de celui de trois occurrences du récepteur liées consécutivement, en s'interrogeant pour chaque occurrence du récepteur membranaire dont les sites r et c sont liés, si elle peut être liée à une même occurrence du récepteur ou si elle peut être liée à deux occurrences différentes. En toute rigueur, pour s'assurer qu'une chaine d'au moins trois occurrences du récepteur ne peut pas se former, il faut également considérer des motifs d'intérêt similaires pour la paire de sites r et n et la paire de sites c et n.

Plus précisément, l'abstraction est paramétrée par le choix d'un ensemble $\mathcal{P}$ de motifs connexes. L'ensemble $\mathcal{P}$ regroupe des motifs d'intérêt, ainsi que des motifs qui seront utilisés de manière intermédiaire dans la preuve que certains de ces motifs d'intérêt sont inaccessibles. Une sous-partie de l'ensemble $\mathcal{P}$ est appelée une propriété abstraite. Chaque propriété abstraite représente un ensemble d'états concrets : un état concret q sera dît compatible avec une propriété abstraite $X^\sharp$ si et seulement si aucun motif qui est dans l'ensemble $\mathcal{P}$ sans être

dans l'ensemble $X^\sharp$ n'apparait dans la configuration d'une espèce biochimique présente dans l'état q. L'ensemble de tous les états concrets compatibles avec la propriété abstraite $X^\sharp$ est alors noté $\gamma_{\mathcal{P}}(X^\sharp)$. Qui peut le plus, peut le moins : plus nombreux sont les motifs autorisés, plus nombreuxs sont les états compatibles. La fonction $\gamma_{\mathcal{P}}$ est donc croissante. Elle permet de définir formellement la notion d'abstraction d'un ensemble d'états : une propriété abstraite $X^\sharp$ sera dite être une abstraction d'un ensemble d'états X si et seulement si l'ensemble X est inclus dans l'ensemble $\gamma_{\mathcal{P}}(X^\sharp)$. La fonction $\gamma_{\mathcal{P}}$ est couramment appelée la *fonction de concrétisation*. De plus l'image d'une propriété abstraite par cette fonction, est appelée sa *concrétisation*.

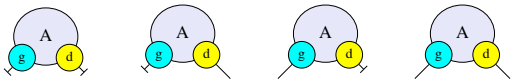
Réciproquement, étant donné un ensemble d'états X, l'ensemble des éléments de l'ensemble $\mathcal{P}$ qui apparaissent dans au moins une configuration d'espèces biochimiques d'un état élément de l'ensemble X sera noté $\alpha_{\mathcal{P}}(X)$. La fonction $\alpha_{\mathcal{P}}(X)$ est croissante également. La propriété abstraite $\alpha_{\mathcal{P}}(X)$ est en fait la *meilleure approximation* de l'ensemble d'états X, ce qui signifie, d'une part, que c'est une abstraction de l'ensemble X (i.e. $X \subseteq \gamma_{\mathcal{P}}(\alpha_{\mathcal{P}}(X))$) et, d'autre part, que c'est un sous-ensemble de toute autre abstraction de X (i.e. pour tout sous-ensemble Y de l'ensemble $\mathcal{P}$ tel que $X \subseteq \gamma_{\mathcal{P}}(Y)$, l'inclusion $\alpha_{\mathcal{P}}(X) \subseteq Y$ est vérifiée). La paire de fonctions $(\alpha_{\mathcal{P}}, \gamma_{\mathcal{P}})$ est alors appelée une *correspondance de Galois* [46, 44].

Exemple 3.2.2 *En figure 3.3 est introduit un exemple jouet pour mieux comprendre le comportement des fonctions d'abstraction et de concrétisation. La signature de ce modèle peut être consultée en figure 3.3(a). Il existe une seule sorte de protéines, qui est appelée A. Cette protéine est munie de deux sites g et d (pour gauche et droite). La carte de contacts spécifie que chaque site peut être libre et qu'un site g peut être lié à un site d d'une même ou d'une autre occurrence de la protéine A. L'abstraction induite par l'ensemble des motifs d'intérêt donné en figure 3.3(b) repose sur les vues locales des occurrences de cette protéine. Elle permet de se poser la question de l'existence ou non, d'une relation entre l'état de liaison des sites g et d dans chaque occurrence de la protéine A.*

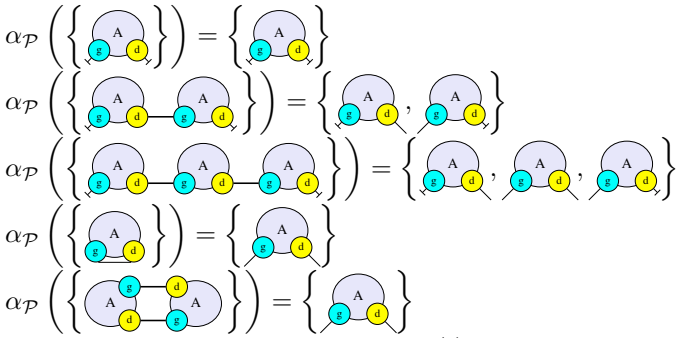
Exemple 3.2.3 *Des exemples de meilleures approximations sont donnés en figure 3.3(c). Par abus de langage, nous appelons la meilleure approximation de la configuration d'une espèce biochimique, la meilleure approximation de l'ensemble formé d'un seul état lui même formé de cette seule configuration d'espèces biochimiques. Le modèle admet deux types de configurations d'espèces biochimiques. Les occurrences de la protéine A peuvent former des chaines d'occurrences de protéines liées successivement par leur site d et g, laissant le site g de la première occurrence de la protéine A et le site d de la dernière occurrence de la protéine A libres. Les occurrences de la protéine A peuvent aussi former des anneaux en reliant le premier et le dernier sites d'une chaine d'occurrences de protéines. La meilleure approximation d'une chaine d'occurrences de la protéine A dépend de la taille de cette chaine. Ainsi, la meilleure approximation d'une chaine réduite à une occurrence de la protéine A est l'ensemble qui contient uniquement la vue locale dont les deux sites sont libres ; la meilleure approximation d'une chaine formée d'exactement deux occurrences de la protéine A est l'ensemble qui contient deux vues locales : l'une avec le site g libre et le site d lié, l'autre avec le site g lié et le site d libre ; enfin la meilleure approximation des chaines d'occurrences de la protéine A de longueur au moins égale à 3 contient également la vue locale dont les deux sites g et d sont liés. Par contre, la meilleure approximation d'un anneau d'occurrences de la protéine A est toujours l'ensemble formé uniquement de la vue locale dont les deux sites g et d sont liées, et ce, quelle que soit la longueur de cette anneau. La fonction qui associe à chaque ensemble d'états sa meilleure approximation est distributive. Cela signifie que la meilleure approximation d'un ensemble d'états est l'union de la meilleure approximation des singletons correspondants.*

Exemple 3.2.4 *Des exemples de concrétisations sont donnés en figure 3.3(d). Par définition, la concrétisation de l'ensemble formé uniquement de la vue locale dans laquelle les deux sites sont libres, est l'ensemble de tous les états qui ne contiennent pas d'autres vues locales. Il s'agit donc des états qui ne contiennent que la configuration de l'espèce biochimique composée d'exactement une occurrence de la protéine A. Par la même démarche, la concrétisation de l'ensemble formé uniquement de la vue locale dont les deux sites sont liés est l'ensemble de tous les états qui ne contiennent que des anneaux d'occurrences de le protéine A (quitte à utiliser cette vue locale plusieurs fois). Par contre, toute configuration d'espèces biochimiques contenant une vue locale avec exactement un site libre, doit contenir également une vue locale avec l'autre site, libre. De ce fait, la concrétisation d'un ensemble composé d'une seule vue locale avec un site libre et l'autre lié, est l'ensemble ne contenant que l'état vide (qui est noté $\emptyset$). Si seules les deux vues locales où un site est lié et l'autre libre sont autorisées, seules des chaines de deux occurrences de la protéine A peuvent être construites. Enfin, si seule la vue locale avec les deux sites libres est interdite, il est possible de former n'importe quelle chaine d'occurrences de la protéine A de taille au moins égale à 2 et n'importe quel anneau d'occurrences de protéines (sans restriction de taille). La*

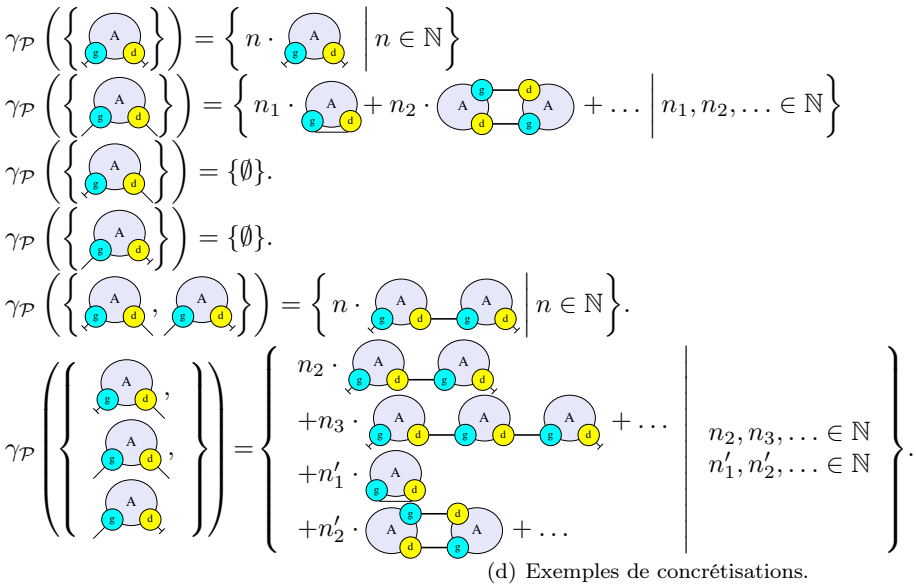
(a) Carte de contacts.



(b) Motifs d'intérêt.

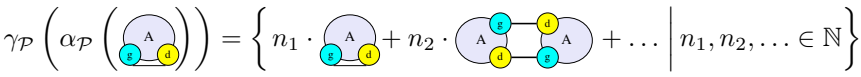


(c) Exemples de meilleures approximations.

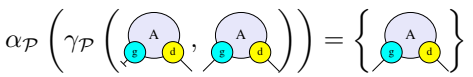


(d) Exemples de concrétisations.

Figure 3.3: Un exemple jouet pour mieux comprendre le comportement des fonctions d'abstraction et de concrétisation. En 3.3(a), la signature du modèle : une seule sorte de protéines, A, avec deux sites pouvant être libres ou liés à l'autre site de la même ou d'une autre occurrence de la protéine A. En 3.3(b), le domaine abstrait est formé des vues locales de l'unique sorte de protéines : toutes les configurations pour les occurrences de la protéine A sont considérées selon que chaque site soit libre ou lié. En 3.3(c) sont donnés des exemples de meilleure approximation d'ensemble d'états. Cela consiste à collecter les vues locales qui peuvent apparaître dans ces états. Réciproquement, En 3.3(d) sont donnés des exemples de concrétisations d'ensembles de vues locales. Ceci consiste à recomposer l'ensemble des états qui ne contiennent aucune occurrence des vues locales manquantes. Dans le cas particulier des vues locales, cela revient à prendre en compte tous les états composés uniquement des vues locales mises à disposition, sachant que chaque vue peut être utilisée zéro, une ou plusieurs fois.



(a) Exemple d'application de la fonction $\gamma_{\mathcal{P}} \circ \alpha_{\mathcal{P}}$.



(b) Exemple d'application de la fonction $\alpha_{\mathcal{P}} \circ \gamma_{\mathcal{P}}$.

Figure 3.4: Suite de l'exemple donné en figure 3.3. Un exemple d'application de la composée de fonctions $\gamma_{\mathcal{P}} \circ \alpha_{\mathcal{P}}$ est montré en 3.4(a). Celui-ci montre que l'abstraction ne permet pas de distinguer des ensembles d'anneaux d'occurrences de la protéine A et ce quels que soient leurs tailles et leurs nombres. En 3.4(b) donne un exemple d'application de la composée de fonctions $\alpha_{\mathcal{P}} \circ \gamma_{\mathcal{P}}$. Cette fonction calcule que la vue locale avec le site g libre et le site d lié ne peut pas apparaître dans une espèce biochimique qui ne contiendrait pas la vue avec le site g lié et le site d libre.

fonction de concrétisation n'est pas distributive (l'image de l'union de deux ensembles de vues locales peut être un sur-ensemble strict de l'union de leurs images).

Les fonctions $\alpha_{\mathcal{P}}$ et $\gamma_{\mathcal{P}}$ se composent dans les deux sens. Ces compositions sont révélatrices des traits principaux du choix de l'abstraction. La composée $\gamma_{\mathcal{P}} \circ \alpha_{\mathcal{P}}$ caractérise le niveau d'approximation. En effet, pour tout ensemble d'états X, $\gamma_{\mathcal{P}}(\alpha_{\mathcal{P}}(X))$ est le plus grand ensemble d'états qui a la même meilleure approximation que X. Il est impossible ainsi de distinguer ces deux ensembles en terme de propriétés abstraites. En revanche, la composée $\alpha_{\mathcal{P}} \circ \gamma_{\mathcal{P}}$ témoigne d'une certaine combinatoire dans le domaine abstrait. Elle associe à chaque propriété abstraite, la plus petite propriété abstraite qui est satisfaite par le même ensemble d'états concrets. Appliquer cette composée permet donc de raffiner une propriété abstraite, par déduction, et ce sans perdre le moindre état concret.

Exemple 3.2.5 *Appliquée à l'ensemble formé d'un seul état composé uniquement d'un anneau de taille* 1*, la composée de fonctions $\gamma_{\mathcal{P}} \circ \alpha_{\mathcal{P}}$ donne l'ensemble des états formés uniquement d'anneaux d'occurrences de la protéine A. En effet, la meilleure approximation d'un anneau de taille* 1*, est l'ensemble de vues locales composé uniquement de la vue dont les deux sites sont liés. Or, voir également en figure 3.3(d), la concrétisation de cet ensemble de vues locales est l'ensemble de tous les états formés uniquement d'anneaux. Ainsi le niveau d'abstraction ne permet de distinguer, ni le nombre d'occurrences, ni la taille des anneaux d'occurrences de la protéine A.*

Exemple 3.2.6 *Appliquée à l'ensemble formé exactement des deux vues locales, la première avec le site g libre et le site d lié, la seconde avec les deux sites liés, la composée de fonctions $\alpha_{\mathcal{P}} \circ \gamma_{\mathcal{P}}$ retourne l'ensemble formé d'une seule vue locale, celle avec les deux sites liés. En effet, la première vue ne peut apparaître dans un état sans que celui-ci ne contienne une occurrence de la vue locale avec le site d libre et le site g lié. De ce fait, elle ne peut apparaître dans aucun état de la concrétisation de l'ensemble formé par ces deux vues locales et n'est donc pas un élément de la meilleure approximation de l'ensemble de ces états. Ainsi, un état abstrait peut contenir des motifs d'intérêt, qui ne peuvent apparaître dans aucune configuration d'espèces biochimiques sans contenir des occurrences de motifs d'intérêt interdits. Retirer ces motifs ne change pas l'ensemble des états concrets qui satisfont la propriété abstraite, mais cette étape peut requérir un temps de calcul substantiel.*

3.3 Transferts de point-fixes

Le plus petit point fixe qui définit l'ensemble des configurations d'espèces biochimiques accessibles dans un réseau réactionnel, pour un état initial donné, peut se calculer au niveau des propriétés abstraites grâce à la correspondance de Galois ($\alpha_{\mathcal{P}}$,$\gamma_{\mathcal{P}}$).

Pour cela, il faut tout d'abord construire la *contre-partie abstraite* de la fonction $\mathbb{F}$, qui agira, non pas sur des ensembles d'états concrets, mais directement sur les propriétés abstraites. Cette contre-partie abstraite se définit de manière systématique : il suffit, pour chaque propriété abstraite $X^\sharp$, de considérer l'ensemble des états concrets $\gamma_{\mathcal{P}}(X^\sharp)$ qui vérifient la propriété $X^\sharp$, puis d'appliquer la fonction $\mathbb{F}$ à cet ensemble et enfin d'appliquer à ce résultat la fonction $\alpha_{\mathcal{P}}$ pour en calculer la meilleure approximation. C'est même la manière correcte la

plus précise de procéder : la fonction $\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}}$ est, en effet, la meilleure contre-partie abstraite de la fonction $\mathbb{F}$ [47]. Elle permet de déléguer le calcul des états accessibles au domaine abstrait en contre-partie d'une perte éventuelle de précision. Pour ce faire, il suffit de remarquer que la fonction $\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}}$ est croissante (comme composée de fonctions croissantes) et définie sur l'ensemble des parties d'un ensemble. Elle admet donc un plus petit point fixe qui sera noté $lfp\ (\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}})$. L'inclusion suivante : $lfp\ \mathbb{F} \subseteq \gamma_{\mathcal{P}}(lfp\ (\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}}))$ se prouve alors par induction [47]. Autrement dit le plus petit point fixe de la fonction $\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}}$ est une abstraction de l'ensemble des états accessibles du modèle considéré. C'est à dire que la propriété abstraite $lfp\ (\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}})$ est satisfaite par chaque état accessible du modèle.

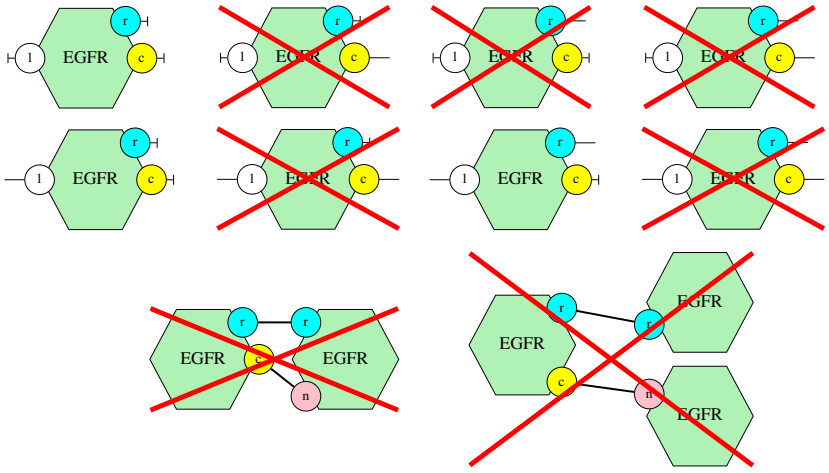
Le calcul des itérations de la fonction $[\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}}]$ peut prendre beaucoup de temps. Il est possible d'ajuster le compromis entre précision et temps de calcul en remplaçant celle-ci par une fonction moins précise. En effet, pour toute fonction croissante $\mathbb{F}^\sharp$ telle que $[\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}}](Y) \subseteq \mathbb{F}^\sharp(Y)$ pour tout ensemble de motifs $Y \subseteq \mathcal{P}$, l'inclusion $lfp\ \mathbb{F} \subseteq \gamma_{\mathcal{P}}(lfp\ (\mathbb{F}^\sharp))$ est également satisfaite [47].

Une telle fonction $\mathbb{F}^\sharp$ peut être dérivée à la main. Pour cela, il faut d'abord donner une définition plus explicite de la fonction $[\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}}]$. Appliquée à un sous-ensemble $Y \subseteq \mathcal{P}$ de motifs d'intérêt, cette fonction ajoute l'ensemble des nouveaux motifs d'intérêt qui peuvent apparaître dans un état accessible en une étape de réécriture à partir d'un état qui ne contient aucun motif de l'ensemble $\mathcal{P}$ qui ne serait pas dans l'ensemble de motifs Y. Or, une telle étape de réécriture est nécessairement induite par une règle-réaction, elle-même induite par une règle du modèle. Chaque nouveau motif P doit donc apparaître dans le membre droit d'une règle-réaction et l'ensemble d'états singleton formé du membre gauche de cette règle-réaction ne doit contenir aucune occurrence de motifs de l'ensemble $\mathcal{P} \setminus Y$. Dans cette règle-réaction, l'occurrence du motif P dont il est question et l'image du membre droit de la règle sous-jacente ont nécessairement au moins une occurrence de protéines en commun (sinon le motif P apparaitrait également dans le membre gauche de la règle-réaction et ne serait dont pas un nouveau motif). Il est alors possible de fixer le motif P au membre droit de cette règle, en unifiant les occurrences de protéines du motif P et de la règle du modèle qui sont communes dans la règle-réaction. Ceci forme alors un raffinement du membre droit de la règle. Un raffinement de la règle peut alors être construit en ajoutant toute information présente dans le motif P qui n'est pas déjà présente dans le membre droit initial de la règle, dans le membre gauche de la règle. Le résultat est une spécialisation de la règle à la production du motif P à cette position particulière. Par construction, le membre gauche de la règle raffinée apparait dans un état dans la concrétisation de Y.

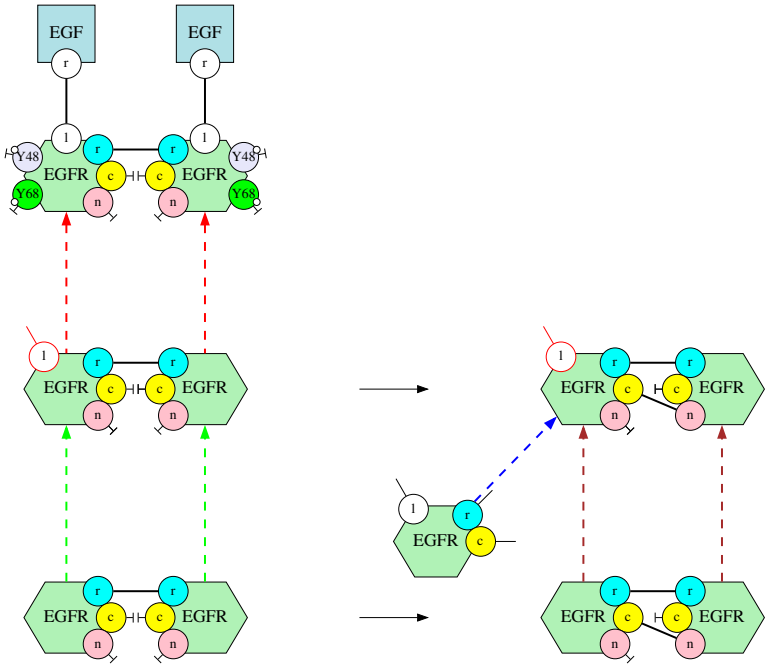
Ainsi, pour calculer les nouveaux motifs d'intérêt de l'ensemble $[\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}}](Y)$, il suffit de calculer tous les *chevauchements* possibles entre un nouveau motif d'intérêt potentiel (dans $\mathcal{P} \setminus Y$) et un membre droit d'une règle du modèle. Chaque chevauchement induit un raffinement de la règle correspondante. Si le membre gauche de la règle raffinée apparait dans un état de l'ensemble $\gamma_{\mathcal{P}}(Y)$, alors ce motif appartient bien à l'ensemble $[\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}}]Y$.

Exemple 3.3.1 *Un exemple de cette construction est dessiné en figure 3.5. L'état abstrait actuel est donné en figure 3.5(a). Seules trois vues locales sont pour l'instant autorisées, celle avec aucun des sites l, r, c lié, celle avec seul le site l lié et celle avec seuls les sites l et r liés. Par ailleurs, ni les doubles liaisons, ni les chaînes d'au moins trois récepteurs ne sont autorisées à cet instant de l'analyse. La preuve que l'on peut construire une occurrence de protéines avec les trois sites l, r et c liés, est donnée en figure 3.5(b). Il suffit d'identifier cette vue locale à l'occurrence gauche du récepteur dans le membre droit de la règle (les plongements en pointillés bleus et marrons envoient ces deux occurrences de protéines sur une même occurrence de protéines) qui permet d'établir une liaison asymétrique entre deux récepteurs membranaires (voir en figure 2.8(e)). Ceci est possible car les sites en commun dans ces deux occurrences de protéines sont dans un état compatible : en effet, la vue locale demande que ces sites soient liés, alors que le membre droit de la règle précise à quels sites ils le sont. La règle est alors spécialisée à la production de la nouvelle vue locale pour ce chevauchement particulier entre la vue locale et le membre droit de la règle. Cela consiste à ajouter dans les membres gauche et droit de la règle l'information que le site l est lié. Pour conclure, il suffit alors d'exhiber un plongement (ici dessiné en pointillés rouges) en le membre gauche de la règle ainsi raffinée et le configuration d'une espèce biochimique, en vérifiant que celle-ci ne contient aucune occurrence des motifs d'intérêt non encore autorisés. Le dimère avec uniquement une liaison symétrique et dans lequel les deux occurrences du récepteur membranaire sont toutes deux liées à des occurrences du ligand et les sites Y48 et Y68 non phosphorylés et libres rempli parfaitement ces conditions. Ainsi la vue locale dans laquelle les trois sites l, r et c sont liés simultanément sera utilisable dès la prochaine itération de l'analyse.*

Lors du calcul de l'ensemble $[\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}}](Y)$, l'étape la plus coûteuse en temps de calcul est de vérifier



(a) Un état abstrait.



(b) La construction de la vue locale dans laquelle les trois sites sont liés peut être construite en une étape à partir de cet état abstrait.

Figure 3.5: Découverte d'un nouveau motif d'intérêt accessible dans le modèle des premières étapes de la voie de l'acquisition du facteur de croissance de d'épiderme (voir en figure 2.8 page 20). En 3.5(a), il est supposé qu'à ce moment de l'analyse, seules trois vues locales sont autorisées. Par ailleurs, il n'est permis de former ni des doubles liaisons entre récepteurs membranaires, ni des chaînes de trois récepteurs ou plus. En 3.5(b), la preuve que la vue locale doit être déclarée accessible à ce niveau d'abstraction est représentée sous forme de diagramme. Elle consiste à appliquer la règle de liaison asymétrique en identifiant la vue locale au récepteur de gauche dans le membre droit de la règle et en raffinant la règle en conséquence. Le membre gauche de la règle obtenue apparaît dans une configuration d'espèces biochimiques ne contenant aucun motif d'intérêt non encore découvert, ce qui conclut la preuve.

que les membres gauches des règles raffinées peuvent apparaître dans un état de l'ensemble $\gamma_{\mathcal{P}}(Y)$. La section suivante a pour but de réduire ce coût moyennant une approximation supplémentaire.

3.4 Analyse par ensembles de motifs orthogonaux

Ajouter des hypothèses sur l'ensemble des motifs d'intérêt et simplifier le test de réalisabilité du membre gauche des raffinements de règles en le remplaçant par une condition nécessaire, mais pas toujours suffisante, permet de rendre ce calcul plus efficace au prix d'une perte de précision de l'analyse. Ceci permet de définir une approximation correcte de la fonction $\alpha_{\mathcal{P}} \circ \mathbb{F} \circ \gamma_{\mathcal{P}}$.

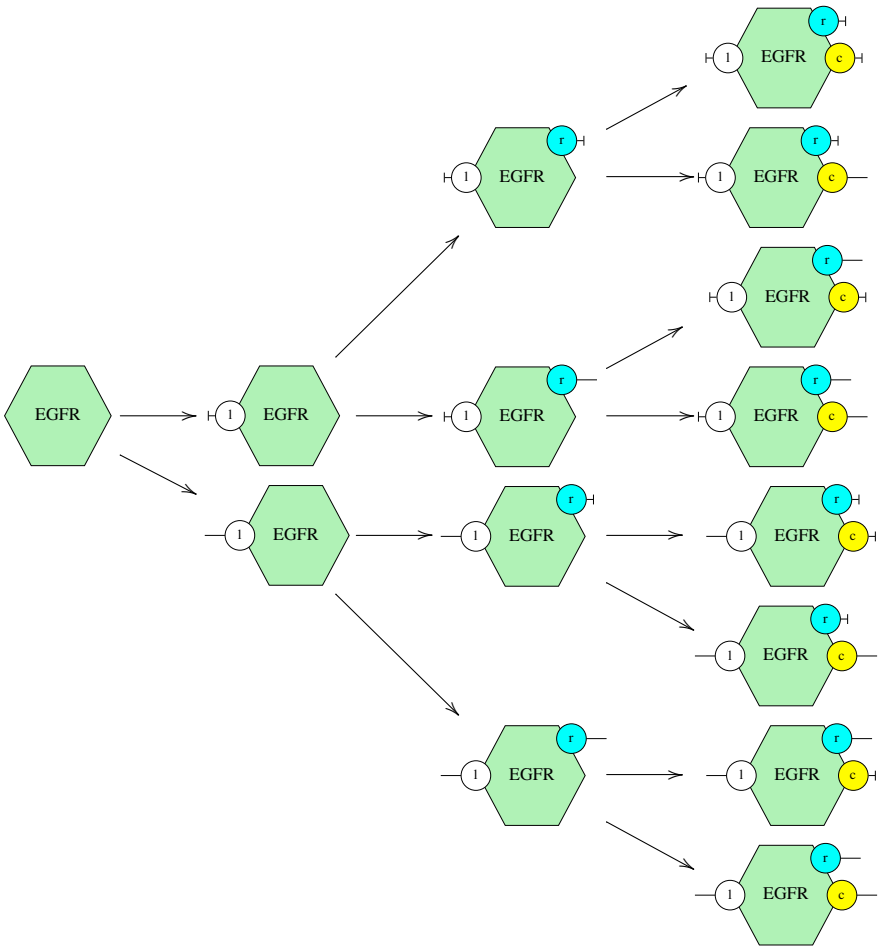
Pour ce faire, l'ensemble des motifs d'intérêt peut être organisé sous la forme d'un ensemble fini d'ensembles finis de motifs orthogonaux [79]. Chaque *ensemble de motifs orthogonaux* est un arbre de décision raffinant progressivement un motif initial, dans le but de répondre à une question spécifique. Un ensemble de motifs orthogonaux est construit de manière à ce que toute occurrence du motif initial dans une configuration d'espèce biochimiques, puisse être complétée en exactement une occurrence d'un de ces raffinements. En conséquence, les raffinements du motif initial sont deux à deux incompatibles et ils recouvrent, en quelque sorte, tous les cas possibles pour le motif initial.

Le choix exact des ensembles de motifs orthogonaux repose sur une analyse préliminaire qui calcule, par inspection des règles du modèle, quelles questions intéressantes se posent. Trois catégories de questions sont considérées par défaut dans l'analyseur KaSa (mais il est possible de paramétrer l'analyse pour en désactiver une ou deux). La première infère des relations entre les états des différents sites de chaque sorte de protéines, cela correspond à l'analyse des vues locales [56]. La seconde permet de détecter des relations entre l'état des sites dans des occurrences de protéines qui partagent un lien [79] dans le but d'analyser le déplacement des occurrences des espèces biochimiques lorsque celui-ci est codé par des transformations de l'état d'activation de sites fictifs. L'analyse permet alors de vérifier si oui ou non deux occurrences de protéines sont toujours localisées dans le même compartiment quand elles sont liées entre elles. La troisième permet de détecter si une même occurrence de protéines peut être liée simultanément à deux occurrences différentes de protéines ou si une même occurrence de protéines peut être liée au moins doublement à une autre occurrence de protéines [79]. Une quatrième sorte d'ensembles de motifs orthogonaux est en cours d'implantation. Elle se concentre sur la formation des espèces biochimiques cycliques : son but est de prouver l'absence de espèces biochimiques de taille non bornée [18].

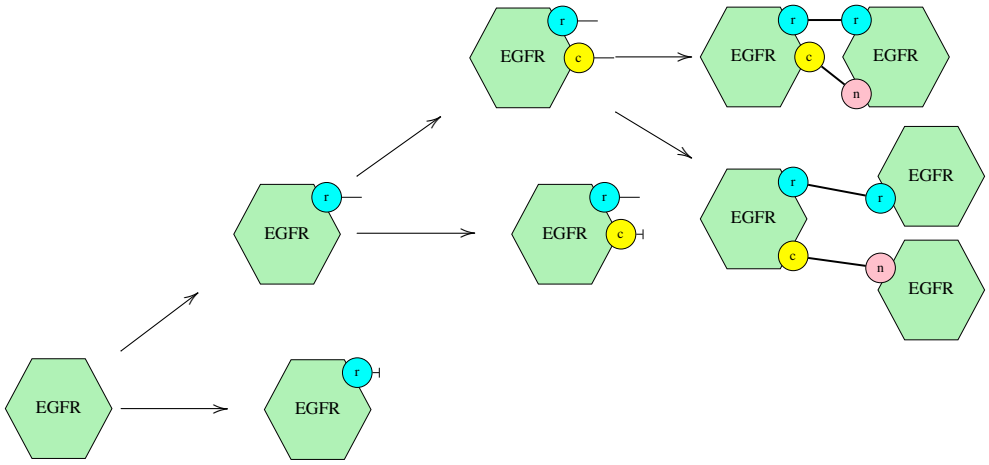
Les ensembles finis de motifs orthogonaux peuvent être construits récursivement, en remplaçant un des motifs par plusieurs motifs le raffinant. Il suffit de choisir une information non spécifiée dans ce motif et de considérer tous les cas possibles pour cette information, d'où la représentation sous forme d'arbre de décision. L'ensemble de motifs orthogonaux est alors formé par les feuilles de cet arbre, alors que les nœuds de cet arbre représentent les motifs intermédiaires qui ont été remplacés par des motifs plus précis.

Exemple 3.4.1 *L'ensemble des motifs d'intérêt introduit en figure 3.2 est inclus dans la réunion de deux ensembles de motifs orthogonaux. En effet, l'ensemble des vues locales peut être obtenu, en partant d'une occurrence de la protéine EGFR sans aucun site, en se demandant successivement si le site l est libre ou non, si le site r est libre ou non et si le site c est libre ou non. L'arbre de décision correspondant se trouve en figure 3.6(a). Les deux derniers motifs d'intérêt sont obtenus en se demandant si un récepteur peut établir des liaisons doubles. Partant d'une occurrence de la protéine EGFR sans aucun site, il faut se demander si le site r est libre ou non, puis dans le cas où le site r est lié, si le site c est lié ou non, et enfin, dans le cas où le site c est également lié, si ces deux sites sont liés à une même occurrence de récepteur membranaire ou à deux occurrences différentes. L'arbre de décision ainsi obtenu est donné en figure 3.6(b).*

Les différents ensembles de motifs orthogonaux collaborent au sein de l'analyse, qui effectue ainsi une induction mutuelle sur ces derniers. Ceci présente deux avantages par rapport à des analyses séparées ou en cascades (où chacune utiliserait le résultat des analyses précédentes). D'une part, il n'est pas nécessaire de définir quel ensemble de motifs orthogonaux doit être analysé avant quel autre. D'autre part, une induction mutuelle est strictement plus expressive. La collaboration entre les différents ensembles de motifs orthogonaux permet de prouver plus souvent que le membre gauche des règles raffinées n'est pas réalisable étant donné les motifs qui sont autorisés à un moment donné de l'analyse (voir en figure 3.5), et donc que la règle peut être ignorée à ce moment de l'analyse. Pour faire cette preuve, le raffinement d'une règle est construit de la manière habituelle. Il suffit ensuite de trouver une occurrence de protéines dans le membre gauche de la règle raffinée qui soit incompatible avec l'état actuel de l'analyse sur au moins un des ensembles de motifs orthogonaux pris



(a) Ensemble de motifs orthogonaux pour les vues locales.



(b) Ensemble de motifs orthogonaux pour discuter des doubles liaisons.

Figure 3.6: Deux exemples d'ensemble de motifs orthogonaux. En 3.6(a), l'ensemble des vues locales (voir les huit premiers motifs en figure 3.2 page 29) sous forme d'arbre de décision. En 3.6(b), celui pour discuter de la présence potentielle de doubles liaisons entre des récepteurs et de la présence potentielle de trimers (voir les deux derniers motifs en figure 3.2).

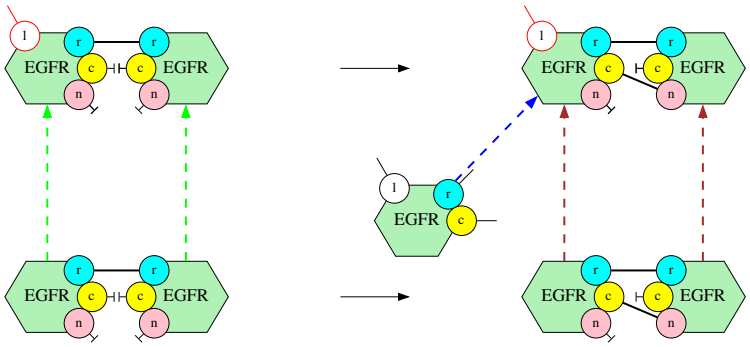


Figure 3.7: L'étape abstraite de la figure 3.5 revisitée avec la procédure de décision approchée. Au lieu de vérifier que chaque motif connexe du membre gauche de la règle raffinée se plonge dans la configuration d'une espèce biochimique dans la concrétisation de l'état abstrait, il suffit de s'assurer, pour chaque occurrence de protéines dans ce motif et chaque ensemble de motifs orthogonaux portant sur ce sorte de protéines si il contient un motif compatible déjà découvert par l'analyse.

en paramètre de l'analyse. Pour cela, la racine de l'ensemble de motifs orthogonaux doit être de la même sorte que l'occurrence de la protéine en question et l'information contextuelle de cette occurrence de protéines dans ce membre gauche de la règle raffinée ne doit être compatible avec aucun des motifs de cet ensemble de motifs orthogonaux déjà déjà déclarés potentiellement accessibles par l'analyse. Dans le cas contraire, l'analyseur ne peut pas prouver que le motif est inaccessible. Le motif est alors considéré comme potentiellement accessible pour la suite de l'analyse. Il s'agit bien entendu d'une approximation.

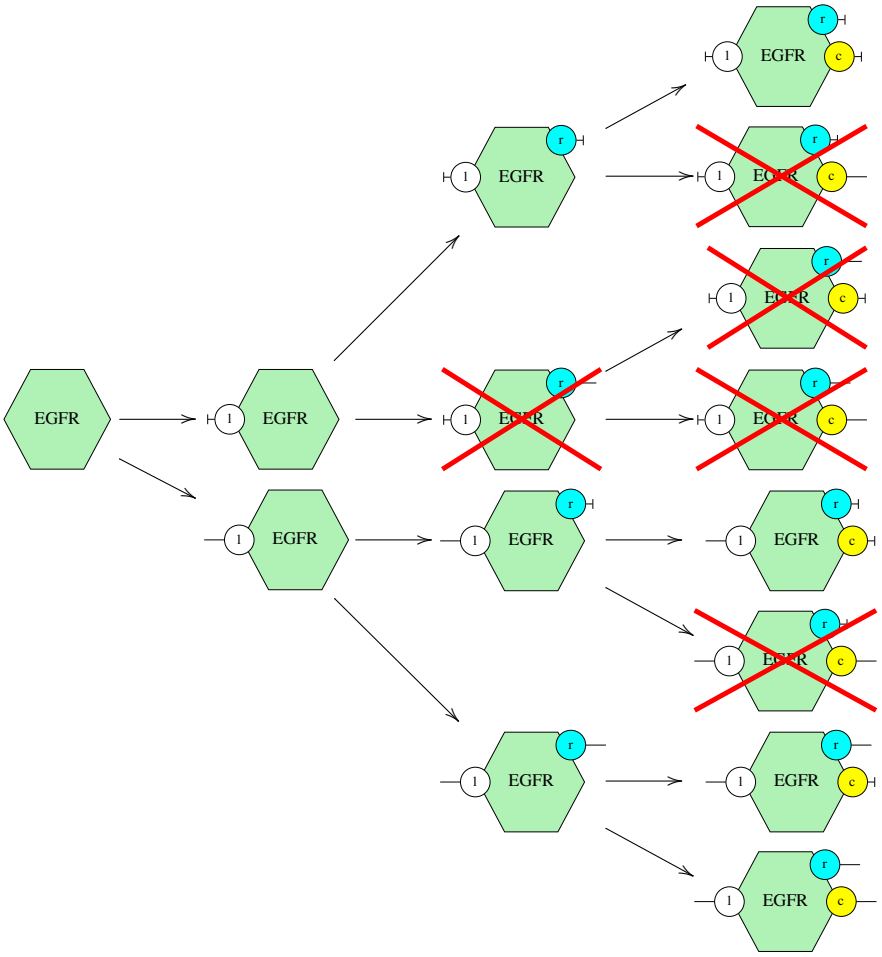
Exemple 3.4.2 *En figure 3.7, l'étape de calcul qui avait été décrite en figure 3.5 est rejouée en remplaçant le test de réalisabilité par cette procédure approchée. Au lieu de construire un plongement du membre gauche de la règle raffinée vers la configuration d'une espèce biochimique afin de vérifier qu'il ne contient pas de motif non encore autorisé, la nouvelle procédure se contente de vérifier pour chaque occurrence de protéines dans le membre gauche de la règle raffinée et pour chaque ensemble de motifs orthogonaux portant sur cette sorte de protéines si celui-ci contient un motif autorisé compatible avec cette occurrence. Dans ce cas, cela revient à vérifier qu'il existe bien une vue locale déjà autorisée dans laquelle les deux sites l et r sont liés, alors que le site c est libre et que le motif dans lequel le site r est lié et le site c est libre est autorisé dans le deuxième ensemble de motifs orthogonaux.*

Outre le fait de ne pas vérifier l'existence de la configuration d'une espèce biochimique qui pourrait compléter le collage obtenu entre les motifs connexes du membre gauche de la règle raffinée et les motifs déjà déclarés potentiellement accessibles par l'analyse, il est intéressant de remarquer que la procédure de décision approchée évite le calcul de tous les chevauchements entre les motifs d'intérêt non encore découverts par l'analyse, en se focalisant sur la racine de chaque ensemble de motifs orthogonaux. Ce sont les deux sources de pertes d'information dues à l'affaiblissement de la procédure de décision.

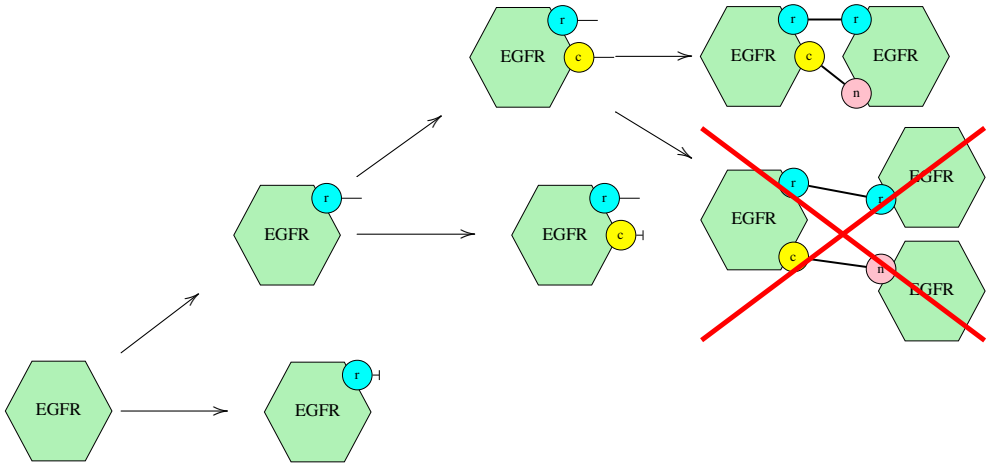
Exemple 3.4.3 *Le résultat de l'itération pour le modèle formé des règles qui avaient été décrites en figure 2.8 pour les ensembles de motifs orthogonaux qui avaient été introduits en figure 3.6, est donné en figure 3.8. Cette itération a été initialisée avec une quantité arbitraire d'occurrences de protéines de chaque sorte, mais avec tous leurs sites libres. Pour ce qui est des vues locales (voir en figure 3.8(a)), seules 4 configurations sont possibles pour l'état des sites l, r, et c des récepteurs membranaires. Ainsi, le site c ne peut être lié sans que le site r ne le soit et le site r ne peut être lié sans que le site l ne le soit. De son côté, l'analyse des doubles liaisons (voir en figure 3.8(b)) montre qu'il est impossible de former des chaînes d'au moins trois récepteurs membranaires.*

Il est important de rappeler que l'analyse ne donne qu'une sur-approximation des états accessibles. Ainsi, tout motif prouvé comme non accessible est bien inaccessible. Par contre, il n'y a aucune garantie qu'un motif non prouvé inaccessible puisse apparaître dans un état accessible depuis un des états initiaux.

3.5 Pour aller plus loin



(a) Analyse d'accessibilité pour l'ensemble des vues locales (voir en figure 3.6(a)).

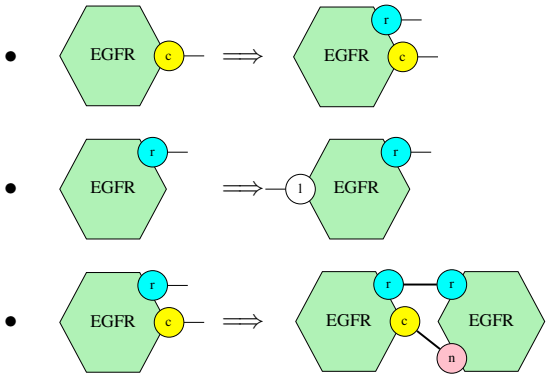


(b) Analyse d'accessibilité pour discuter de la présence éventuelle de trimers et de double liaisons.

Figure 3.8: Résultat de l'analyse pour les deux ensembles de motifs orthogonaux donnés en figure 3.6. Les motifs orthogonaux sont aux feuilles des arbres de décision. Ceux qui sont barrés en rouge n'apparaissent dans aucune exécution du modèle (pour n'importe quel état initial sans lien). Par construction de l'arbre de décision, les nœuds dont tous les enfants sont inaccessibles sont également inaccessibles et donc barrés eux-aussi.

L'itération de point-fixe est suivie d'une phase de traitement du résultat. Le but est essentiellement de rendre le résultat de l'analyse plus compréhensible pour l'utilisateur. Dans un premier temps, un parcours de chaque arbre de décision est effectué et chaque nœud dont tous les fils sont déclarés inaccessibles est déclaré inaccessible lui-aussi. Ensuite, tous les nœuds des arbres de décision sont explorés en répertoriant ceux dont les enfants n'ont pas tous le même statut. Ceci témoigne d'une propriété intéressante puisque dans ce cas, un des raffinements d'un motif accessible n'est pas accessible. Cette information est alors présentée sous la forme d'une implication, appelée *lemme de raffinement*, entre un motif (le nœud en question) et une liste de motifs (ses fils qui n'ont pas été prouvés inaccessibles). Une telle implication s'interprète de la manière suivante : chaque occurrence du motif de la précondition dans une configuration accessible d'une espèce biochimique peut toujours se raffiner en l'un des motifs de la postcondition.

Exemple 3.5.1 *Le résultat de l'analyse décrit en figure 3.8 donne lieu aux implications suivantes :*



Cela prouve que dans une occurrence du récepteur membranaire, le site c ne peut être lié sans que le site r ne le soit également, et que le site r ne peut être lié sans que le site l ne le soit aussi. De plus, une occurrence du récepteur dont les sites r et c sont tous deux liés, est nécessairement liée doublement à une même occurrence du récepteur.

Par ailleurs, l'analyseur vérifie pour chaque règle si son membre gauche est compatible avec le résultat de l'analyse (avec la procédure de décision simplifiée présentée Sec. 3.4). Les règles pour lesquelles ce n'est pas le cas sont reportées à l'utilisateur.

Nous avons utilisé l'analyseur statique KaSa sur plusieurs modèles. Les résultats de ces analyses sont décrites en table 3.1. Les onze premiers modèles sont des traductions directes des modèles qui sont fournis avec la distribution du logiciel BNGL [9]. Le modèle 'egfr' décrit les premiers événements de l'acquisition du facteur de croissance de l'épiderme (il comprend entre autres les règles données en figure 2.8). Le modèle 'egfr, erk, mapk, ras' va beaucoup loin dans la voie de signalisation [51, 10, 124, 20]. Les modèles 'machine' et 'ensemble' sont deux versions de la voie de signalisation MAPK, publiées par Eric Deeds et Ryan Suderman [126]. Les versions du modèle 'korkut' concernent la voie de signalisation de la protéine Ras. Ce modèle a été conçu par John Bachman et Benjamin Gyori (Sorger lab) dans le cadre du projet DARPA Big Mechanism [38] en utilisant des outils de traitement automatique du langage naturel pour extraire des faits de la littérature, en assemblant ces faits en Kappa, puis en corrigeant manuellement les modèles obtenus [89]. Le modèle 'tgf' s'intéresse lui à la matrice extra-cellulaire de la protéine tgf-β. Cinq versions de ce modèle ont été analysées. Elles ont été assemblées à la main d'après la littérature par Nathalie Théret et Jean Coquet, puis corrigées avec l'aide de l'analyseur KaSa. Enfin, plusieurs versions du modèle de la voie de signalisation de la protéine Wnt, écrites par Héctor F. Medina Abarca (Fontana Lab) dans le cadre du projet DARPA Big Mechanism [38] ont été analysées. Ce modèle a également été assemblé manuellement après lecture humaine de la littérature. Dans ce dernier modèle, le grand nombre de règles vient du fait que des scripts ont été utilisés pour raffiner des règles d'interactions génériques afin d'ajuster leurs cinétiques en fonction d'informations contextuelles sur les configuations des occurrences de protéines qui interagissent et de celles de leurs voisines.

Pour chaque modèle et chaque version, nous avons reporté le nombre total de règles, ainsi que nombre d'implications découvertes par l'analyse portant sur des relations soit entre au moins deux sites, soit entre les états de liaison et d'activation d'un même site. Nous avons également donné le nombre de règles qui ont été trouvées mortes par l'analyseur statique (du fait de l'approximation, l'analyseur peut manquer des règles mortes, par contre, toute règle détectée morte l'est). Nous donnons également le temps de calcul total de

modèle	nombre de règles	nombre de contraintes inférées	nombre de règles mortes détectées	temps d'analyse (secondes)
repressilator	42	0	0	0.005
egfr_net	39	4	0	0.011
egfr_net_red	45	6	0	0.013
fceri_fyn	46	4	0	0.025
fceri_fyn_lig	48	4	0	0.025
fceri_fyn_trimer	362	4	36	0.301
fceri_fyn_gamma2	59	5	0	0.036
fceri_fyn_ji	36	4	0	0.019
fceri_fyn_ji_red	32	4	0	0.017
fceri_fyn_lyn_745	40	4	2	0.021
fceri_fyn_trimer	192	4	0	0.136
egfr	20	9	0	0.010
egfr, erk, mapk, ras	69	7	0	0.046
machine	220	13	7	0.405
ensemble	233	26	0	0.364
korkut (2017/01/13)	3916	0	1610	7.92
korkut (2017/01/17)	12896	0	874	24
korkut (2017/02/06)	5750	0	884	57
TGF (V19)	97	19	10	0.223
TGF (V20)	99	30	10	0.275
TGF (V21)	211	18	0	0.652
TGF (2017/04/01)	235	13	0	0.534
TGF (2018/04/19)	292	13	0	0.625
BigWnt (2015/12/28)	356	2	1	2.06
BigWnt (2016/09/28)	1419	1	0	6.64
BigWnt (2017/03/22)	1486	14	12	8.74

Table 3.1: Résultats expérimentaux (calculés sur un MacBook Pro avec une puce Intel Core i7-6567U (cadencée 3.3 GHz)). Pour chaque modèle et chaque version, le nombre de règles est donné, ainsi que le nombre de contraintes découvertes par l'analyse et le nombre de règles mortes trouvées (qui ne sont donc jamais utilisées dans le modèle). Le temps total de l'analyse est également fourni.

l'analyse, ce qui montre que KaSa passe à l'échelle même sur des modèles comportant un grand nombre de règles d'interactions.

Les informations trouvées par l'analyse statique sont utiles. Une même démarche a été suivie pour améliorer la qualité de ces modèles, qu'ils soient écrits à la main ou assemblés automatiquement par fouille automatique de la littérature. La première étape est la vérification des règles mortes. Ces règles sont souvent la conséquence, soit d'erreurs typographiques, soit d'états initiaux incomplets, soit de règles manquantes, soit de relations de causalité qui ne peuvent pas être satisfaites. La lecture des contraintes trouvées par l'analyseur permet de mieux comprendre leurs origines. Elle permet également de vérifier que les invariants structurels auquel le modélisateur peut s'attendre sont bien vérifiés. L'étape suivante est d'étudier comment une occurrence de protéines peut passer d'une configuration à une autre. L'analyse des *traces locales* [77, 78] calcule des systèmes de transitions à partir des vues locales. Ceci permet d'avoir une cartographie des changements de configuration de chaque occurrence de protéines en faisant abstraction de l'état des occurrences de protéines auxquelles cette occurrence est liée. En particulier, une étude de ces systèmes de transitions permet de calculer efficacement des transitions qui sont définitives : c'est à dire celles qui transforment la configuration d'une occurrence de protéines, sans retour possible, quel que soit le nombre de transitions ultérieures.

Exemple 3.5.2 *Le système de transitions qui représente les traces locales des occurrences du récepteur membranaire dans le modèles des premières interactions qui interviennent dans l'activation du facteur de croissance de l'épiderme, est dessiné en figure 3.9. La succession entre les différentes configurations possibles y est clairement décrite. Ainsi, partant d'un récepteur avec les sites l, r, c et n libres (le site n a été ajouté pour rendre*

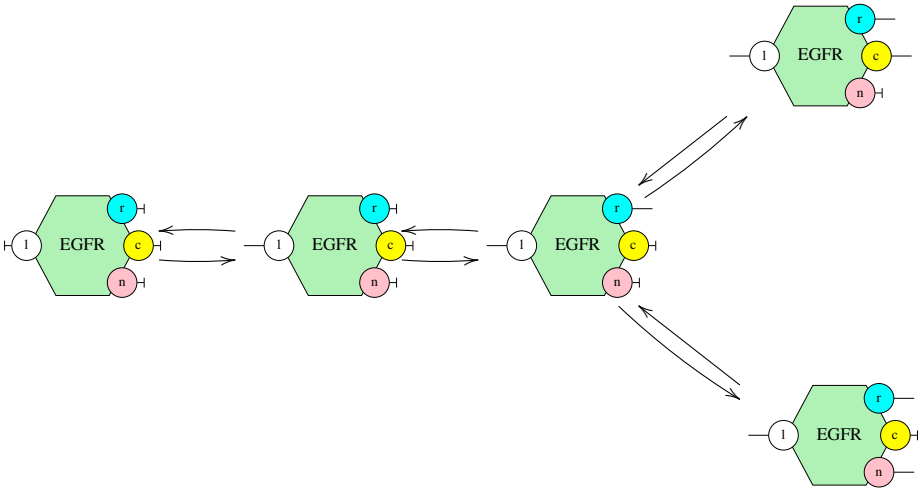


Figure 3.9: Le système de transitions pour les vues locales des occurrences du récepteur membranaire. L'état du site n a été ajouté pour rendre l'exemple plus intéressant. Ce système de transitions explique les étapes que traverse ces occurrences lorsque leurs sites deviennent liés. Les transitions en sens inverse, qui correspondent aux règles de libération des sites sont aussi représentées.

l'exemple plus intéressant), le site l peut devenir lié en premier, ensuite le site r peut devenir lié, ensuite soit le site c, soit le site n peut devenir lié. Par contre, ces deux liaisons sont exclusives : les sites c et n d'une occurrence de protéines ne peuvent pas être liés tous les deux simultanément. Par ailleurs, toutes les liaisons peuvent se défaire dans l'ordre inverse de leur création.

Bien que très efficace, la principale limite de l'analyse d'accessibilité présentée dans ce chapitre est qu'elle reste trop lente pour être utilisée en cours de mise à jour d'un grand modèle dans un éditeur de texte. Dans ce contexte, le calcul des invariants recommence de zéro à chaque modification. Une analyse incrémentale, qui serait capable de tirer profit des analyses précédentes quand une règle est ajoutée ou retirée, ou quand l'état initial est changé, serait plus appropriée. Les ajouts de règles ne posent pas de grandes difficultés. Il suffit de continuer l'itération à partir du point fixe précédent. Par contre, lorsqu'une règle est enlevée, une telle approche conduirait à une trop grande perte de précision, puisque la règle en question est susceptible d'être nécessaire à l'accessibilité de certains motifs, ce qui serait alors impossible d'exploiter pour obtenir un résultat précis. Un algorithme de saturation opérant sur des clauses de Horn permettra de mieux gérer la suppression de règles. Dans cette approche, chaque clause constitue une implication formalisant le fait que s'il est impossible, avec le niveau d'abstraction fixé, de prouver que certains motifs sont innaccessibles et que si une règle donnée existe dans le modèle, alors il est impossible de prouver que tel autre motif est innacessible. En fait, chaque instance du diagramme donné en figure 3.5(a), se traduit en une clause de Horn. En plus, de l'ensemble des motifs dont il a été prouvé qu'il est impossible de prouver l'inaccessibilité, l'analyseur se rappellera quels prédicats et quelles clauses ont été utilisées pour prouver chaque prédicat. Il sera alors plus facile des les invalider de proche en proche lors de la suppression d'une règle avant de reprendre l'algorithme de saturation, pour vérifier s'il n'existe pas d'autres preuves des prédicats ainsi invalidés.

Chapitre 4

Flot d'information dans la sémantique différentielle d'un modèle Kappa

Les sémantiques quantitatives permettent de définir et d'étudier le comportement des modèles au cours du temps. Dans ce chapitre, seule la sémantique différentielle sera considérée. Elle décrit l'évolution des quantités de chaque constituant du modèle sous la forme d'équations différentielles ordinaires. Elle est donc déterministe. La sémantique différentielle demande de préciser la vitesse des réactions biochimiques. Il suffit pour cela d'associer à chaque règle de réécriture une constante pour spécifier la vitesse des réactions induites par ces règles. Toutefois, une telle sémantique ne passe pas à l'échelle des grands systèmes d'interactions, car elle requiert une variable par configuration d'espèces biochimiques. Il est donc nécessaire de proposer des méthodes de réduction pour définir des systèmes différentiels de dimensions plus petites.

La méthode présentée ici s'appuie sur l'analyse du flot d'information entre les différents sites des espèces biochimiques. Elle permet d'identifier des paires de sites dont la corrélation entre les états n'a aucune influence sur la dynamique du système considéré. Chaque configuration d'une espèce biochimique peut alors être séparée en portions plus petites qui se comportent de manière autonome. L'explosion combinatoire due à la taille des espèces biochimiques est alors contournée.

En section 4.1 est illustrée une des raisons principales de l'explosion combinatoire en nombre de configurations potentielles des espèces biochimiques dans les modèles de réécriture de graphes à sites. En section 4.2 les calculs pour réduire la dimension d'un cas d'étude jouet sont effectués à la main. En section 4.3 la notion de réduction de modèle différentiel est formalisée. Ce cadre générique est ensuite appliqué aux modèles écrits en Kappa, en section 4.4.

Remerciement

Ces travaux sont principalement issus de mon post-doc auprès de Walter Fontana, qui ont été publiés dans [74]. La thèse de Ferdinanda Camporesi a permis une formalisation plus complète et une extension dans le cas non-uniforme [23].

4.1 Une brèche dans le mur de la combinatoire

4.1.1 Les causes de l'explosion combinatoire

Le début de ce chapitre explique une des raisons de l'explosion combinatoire dont souffrent les systèmes d'équations différentielles qui émergent des modèles d'interactions biochimiques entre protéines.

En figure 4.1 est dessinée une configuration typique d'une occurrence d'un dimère dans le modèle des premières étapes dans l'acquisition du facteur de croissante de l'épiderme [9]. Dans ce modèle, il est intéressant de mesurer la quantité de la protéine cible *Sos* qui est attachée à la membrane de la cellule. C'est à dire la quantité de cette protéine attachée soit directement à la membrane par le biais d'une occurence de la protéine de transport *Grb2*, soit indirectement par l'intermédiaire d'une occurrence de la protéine d'échafaudage *ShC*. Chaque occurrence d'un dimère contient deux occurrences du récepteur *EGFR* et chaque occurrence du récepteur contient deux sites *Y48* et *Y68* susceptibles de recruter une occurrence de la protéine *Sos*. Les deux

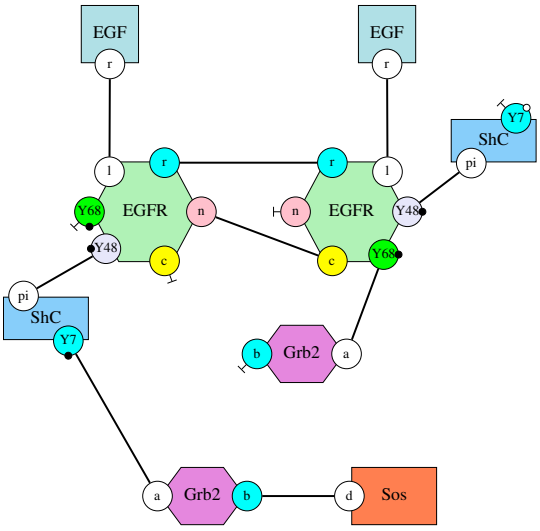


Figure 4.1: Un exemple de configuration pour un dimère. Cette configuration contient deux occurrences du site *Y48* et deux occurrences du site *Y68* chacune suceptible de recruter une occurrence de la protéine *SoS*, d'où l'explosion combinatoire du nombre potentiel de configurations d'espèces biochimiques.

occurrences de la protéine recepteur étant distinguées dans le dimère, il y aura donc $n_{short}^2 \cdot n_{long}^2$ où n_{short} est le nombre d'états intermédiaires dans l'acquisition de la protéine *Sos* par la voie directe et n_{long} est le nombre d'état intermédiaire dans l'acquisition de la protéine *Sos* par la voie indirecte.

Toutefois, plutôt que de mesurer la quantité de chaque configuration d'espèces biochimiques en multipliant par le nombre d'occurrences de la protéine *Sos* que chacune a recruté, chaque occurrence des sites *Y48* et *Y68* peut être vue indépendamment en oubliant qu'ils apparaissent sur la même occurrence d'une configuration de dimères. Ceci revient à considérer la configuration d'un dimère comme un composant qui contient quatre processus, tout en estimant que savoir que ces quatres processus sont sur la même configuration d'un dimère n'est pas une information primordiale. Ces processus auraient tout autant pu se trouver sur des occurrences différentes, l'évolution de la quantité globale de la protéine *Sos* recrutée par la membrane aurait été la même. Ces processus ont alors $2 \cdot (n_{short} + n_{long})$ états différents (le facteur 2 vient du fait que les deux occurrences de la protéine récepteur sont distinguées par leur liaison asymétrique).

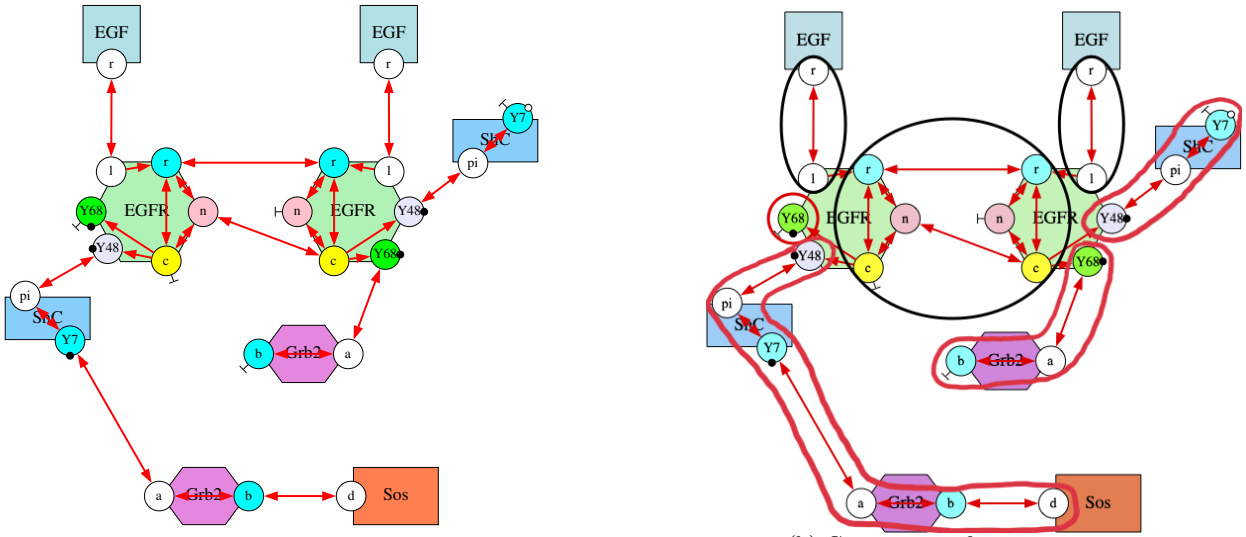
4.1.2 Le flot d'information

Le *flot d'information* entre les sites d'interactions de la configuration d'une espèce biochimique permet d'établir l'indépendance de ces quatre processus. Intuitivement, le flot d'information est une relation entre les sites d'une configuration d'espèces biochimiques qui indique l'état de quels sites est susceptible d'influencer la modification de l'état de quels autres sites. Elle prend la forme d'un graphe dont les nœuds sont les sites d'interactions d'une configuration d'espèces biochimiques et les arcs (orientés) peuvent relier soit deux sites sur une même occurrence de protéines, soit deux sites liés entre eux. Un chemin d'un site source vers un site cible témoigne que l'état du site source influence potentiellement la capacité à modifier l'état du site cible. Par contre, l'absence d'un tel chemin signifie que l'état du site source n'a aucune influence sur la modification éventuelle de l'état du site cible. La notion de flot d'information dans les modèles écrits en Kappa sera formalisée en section 4.4.2.

4.1.3 Réduction de la combinatoire

Une fois annotée par une sur-approximation de son flot d'information, il est possible d'identifier des portions de configurations d'espèces biochimiques dont le comportement peut être décrit de manière indépendante.

Ceci repose sur une décomposition en composantes fortement connexes des graphes formés par le flot d'information entre les sites d'interactions des configurations de chaque espèce biochimique. Il faut ainsi regrouper sur chaque graphe toute paire de sites tels qu'il existe un chemin de l'un vers l'autre et réciproquement. La décomposition du flot d'information dessinée en figure 4.2(a) est donnée en figure 4.2(b). Deux sortes de composantes fortement connexes sont à distinguer. Celles en rouge sont dîtes terminales car il n'est pas possible



(a) Flot d'information.

(b) Composantes fortement connexes.

Figure 4.2: Flot d'information dans la configuration d'un dimère. En 4.2(a), des flèches rouges relient deux sites sur une même occurrence de protéine ou deux sites sur un liaison. Un chemin d'un site source vers un site cible témoigne que l'état du site source influence potentiellement la capacité à modifier l'état du site cible. En 4.2(b), le graphe du flot d'information est décomposé en composantes fortement connexes. Chaque composante fortement connexe est représentée par une forme arrondie. Les composantes fortement connexes terminales sont dessinées en rouge alors que les autres le sont en noir.

d'en sortir en suivant le flot d'information. Les autres sont représentées en noir. Pour obtenir une portion de la configuration d'un dimère dont le comportement peut être décrit indépendamment des autres sites d'interactions de cette configuration, il suffit de compléter chaque composante fortement connexe terminale en suivant le flot d'information en arrière. Ceci donne quatre portions de configuration d'espèces biochimiques, chacune correspondant à l'état d'avancement d'une occurrence du site *Y48* ou *Y68* dans le recrutement d'une occurrence de la protéine *Sos*, ce qui était le but.

Il semble ainsi possible d'exploiter l'absence de flot d'information pour détecter les corrélations inutiles, et ainsi découper les configurations d'espèces biochimiques en portions plus petites qui auront un comportement autonome.

4.2 Exemple jouet

Un exemple jouet permetra de mieux comprendre comment tout ceci fonctionne en pratique. Celui-ci implique une seule sorte de protéine, munie de trois sites : un site en haut, en rouge ; un site gauche, en vert ; et un site droit, en bleu. Chaque site pourra être phosphorylé ou non. Il n'y a pas de liaisons dans ce modèle.

Des hypothèses sont faites en ce qui concerne comment l'état de chaque site influence la modification de l'état des autres sites. Dans chaque occurrence de la protéine, l'état du site rouge contrôle à la fois l'évolution de l'état du site vert et celle du site bleu, comme indiqué dans la carte de contacts annotée en figure 4.4(a). Par contre, l'état du site vert n'a pas d'incidence sur l'évolution de l'état du site bleu, et réciproquement, l'état du site bleu n'a pas d'incidence sur l'évolution de l'état du site vert.

Pour permettre de donner l'expression de toutes les dérivées, seul un ensemble minimal de règles-réactions sera considéré. Elles sont appelées règles-réactions car ce sont des règles Kappa dans lesquelles l'état de tous les sites des protéines qui interagissent est spécifié. Elles peuvent donc être vues comme des réactions entre des configurations d'espèces biochimiques. Ces règles-réactions sont dessinées en figure 4.4(b). La première spécifie que le site du haut, en rouge, peut se faire phosphoryler quand les deux autres sont déphosphorylés. La constante de cette règle-réaction est k^c. Une fois le site du haut phosphorylé, le site gauche, en vert, et le site droit, en bleu, peuvent se faire phosphoryler à leur tour. Le fait que la phosphorylation au préalable du site du haut soit nécessaire à cela justifie le flot d'information du site du haut vers les deux autres sites. Sur la deuxième ligne de règles-réactions, la constante de phosphorylation, k^g, du site gauche, en vert, ne dépend pas

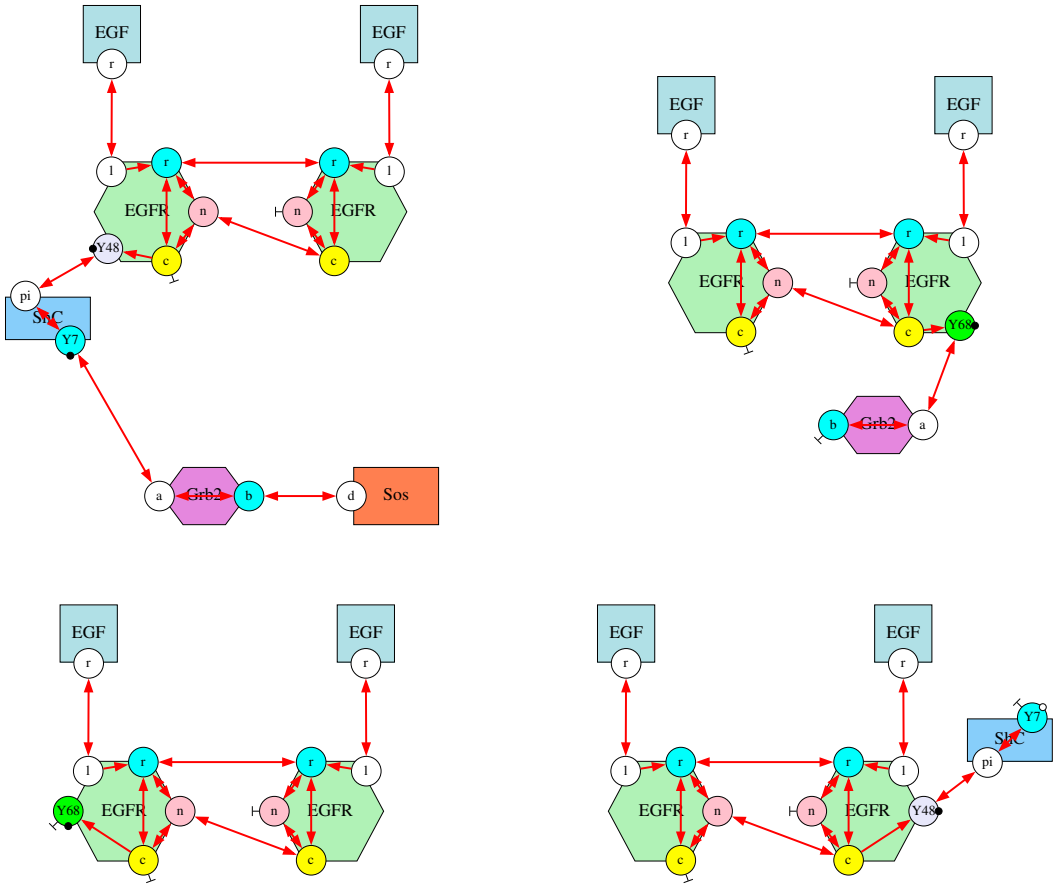
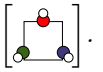


Figure 4.3: Chaque composante fortement connexe terminale est complétée des sites qui ont une influence sur elle. Chacune donne lieu à une portion de dimère qui se comporte de manière indépendante.

du fait que le site droit, en bleu, soit déjà phosphorylé ou non. Il n'y a donc pas de flot d'information du site bleu vers le site vert. De même, sur la troisième ligne de règles-réactions, la constante de phosphorylation, k^d du site droit, en bleu, ne dépend pas du fait que le site gauche, en vert, soit déjà phosphorylé ou non. Il n'y a donc pas de flot d'information entre l'état du site vert vers l'état du site bleu.

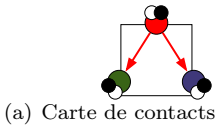
Le comportement de ce modèle est défini par l'application de la *loi d'action de masse*. Chaque réaction (ou ici règle-réaction) induit une contribution au système d'équations différentielles. L'activité de la réaction s'exprime comme le produit de la constante de la réaction et de la quantité des réactifs (qui apparaissent dans le membre gauche de la règle). Chaque réactif est alors consommé proportionnellement à l'activité de la réaction correspondante alors que chaque produit est ajouté dans la même quantité.

Exemple 4.2.1 *La première règle-réaction a pour activité* $k^c \cdot$  *. Elle induit deux termes dans le système d'équations différentielles :*

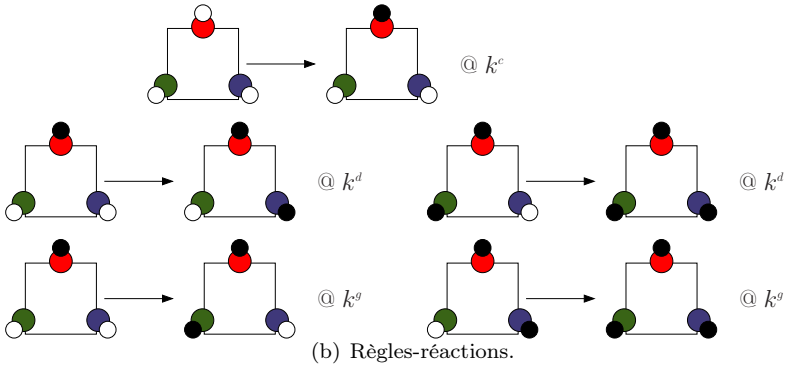
$$\begin{cases} \dfrac{d\left[\quad\right]}{dt} \stackrel{+}{=} -k^c \cdot \left[\quad\right] \\ \dfrac{d\left[\quad\right]}{dt} \stackrel{+}{=} k^c \cdot \left[\quad\right] \end{cases}$$

Le système différentiel complet est donné en figure 4.4(c).

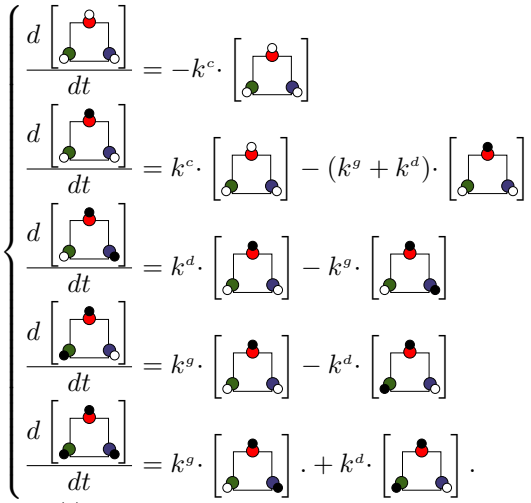
Il existe plusieurs conventions pour tenir compte des symétries éventuelles quand une réaction contient plusieurs occurrences d'un même réactif [15]. Elles prennent alors la forme d'un facteur correctif à appliquer à l'activité des réactions. Ce n'est le cas d'aucune réaction de cet exemple.



(a) Carte de contacts annotée.

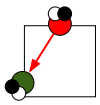


(b) Règles-réactions.

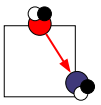


(c) Système d'équations différentielles induit.

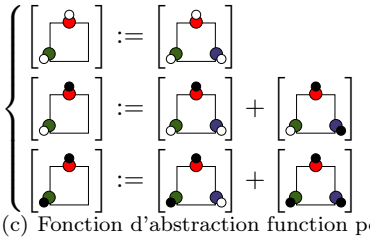
Figure 4.4: Un exemple jouet. Une protéine contient trois sites qui peuvent être phosphorylés ou non. En 4.4(a), la carte de contacts annotée indique que l'état du site du haut, en rouge, peut influencer la modification de l'état du site gauche, en vert, et l'état du site droit, en bleu. Par contre, l'état du site gauche n'a pas d'influence sur la capacité à modifier l'état du site droit, et réciproquement, l'état du site droit n'a pas d'influence sur la capacité à modifier l'état du site gauche. Ceci se traduit au niveau des règles d'interactions (voir en 4.4(b)). Une fois le site du haut phosphorylé (par la première réaction). Le site gauche peut se faire phosphoryler avec la constante de réaction, k^g, que le site droit soit déjà phosphorylé ou non. De même, le site droit peut se faire phosphoryler avec la constante de réaction, k^d, que le site gauche soit déjà phosphorylé ou non. Ce modèle ne contient que des étapes de phosphorylation pour garder le système différentiel de taille raisonnable. En appliquant le principe de la loi d'action de masse, ces règles-réactions induisent le système différentiel donné en 4.4(c).



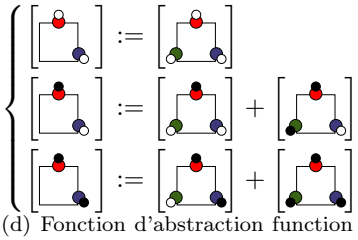
(a) Carte de contacts pour la portion gauche de la protéine.



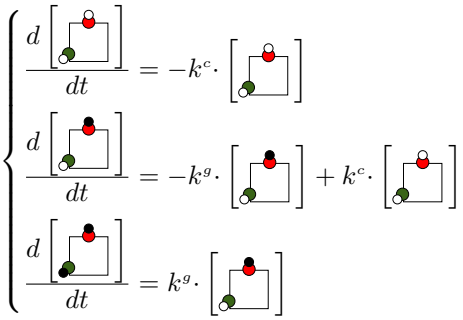
(b) Carte de contacts pour la portion droit de la protéine.



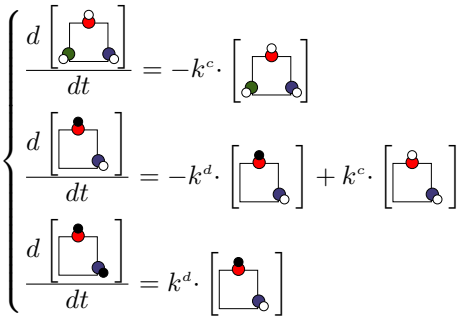
(c) Fonction d'abstraction function pour la portion gauche de la protéine.



(d) Fonction d'abstraction function pour la portion droite de la protéine.



(e) Système d'équations différentielles pour la portion gauche de la protéine.



(f) Système d'équations différentielles pour la portion droite de la protéine.

Figure 4.5: Réduction de la sémantique différentielle de l'exemple jouet (voir en figure 4.4). En utilisant le fait que l'état du site gauche de la protéine n'influence pas l'évolution de l'état du site droit et que réciproquement, l'état du site droit n'influence pas l'état du site gauche, il est possible de découper la protéine en deux parties. La quantité d'un motif est définie comme la combinaison linéaire de la quantité des configurations de la protéine compatible avec ce motif en 4.5(c) et 4.5(d). En 4.5(e) et 4.5(f) sont exprimées les dérivées de la quantité des différents motifs en fonction de la quantité de ces motifs.

Il est possible d'exploiter l'absence de flot d'information du site vert vers le site bleu, et du site bleu vers le site vert. Ceci revient à découper chaque occurrence de protéines en deux portions. Comme l'état du site rouge contrôle à la fois l'évolution de l'état du site vert et celle du site bleu, ces deux portions devront se chevaucher sur le site rouge. Ainsi, les portions gauches de la protéine documenteront l'état du site rouge et du site vert alors que les portions droites documenteront l'état du site rouge et du site bleu. Le flot d'information entre les sites de ces portions de protéines est donné en 4.5(a) et 4.5(b).

Pour utiliser les différentes configurations de portions de la protéine comme des variables, il faut leur donner un sens formel. Celui-ci provient du principe de dualité entre leurs significations intensionnelles et extensionnelles. De manière intensionnelle, la configuration d'une portion de la protéine peut être comprise comme le sous-graphe d'une configuration complète de la protéine. De manière extensionnelle, la configuration d'une portion de la protéine peut être vue comme le multi-ensemble de toutes les configurations complètes de la protéine qui contiennent cette portion, multipliées par le nombre de plongement de cette portion dans chaque configuration complète. Sous cette angle, la quantité la configuration d'une portion de la protéine peut se définir comme la combinaison linéaire des quantités des configurations complètes de la protéine qui contiennent cette portion. Ceci permet d'obtenir les définitions en figures 4.5(c) et 4.5(d). En particulier, la quantité d'une portion de protéine dont aucun site n'est phosphorylé est égal à la quantité des protéines entièrement déphosphorylées (puisque le site rouge doit être phosphorylé en premier). La quantité des autres portions est obtenue en sommant la quantité des configurations obtenues selon l'état du site non documenté.

Les équations qui décrivent l'évolution des quantités décrites en figures 4.5(c) et 4.5(d) peuvent ensuite être dérivées. Elles sont données en figures 4.5(e) et 4.5(f) et peuvent être vérifier analytiquement à partir des équations en figure 4.4(c).

Réduire un système d'équations différentielles à 5 variables en un système à 6 variables n'est guère impressionnant. Toutefois, en regardant de plus près, les 5 variables du système initial correspondent à 1 variable pour la configuration avec le site rouge déphosphorylé et à 2×2 variables pour les autres configurations, selon que le site vert soit phosphorylé ou non, et selon que le site bleu soit phosphorylé ou non. Dans le modèle réduit, la variable pour la configuration avec le site rouge déphosphorylé est représentée de manière redondante par deux variables du système réduit (leur valeur restera donc égale au cours de l'éxecution du système réduit). Les variables pour les autres configurations correspondent à $2+2$ variables, car il y a deux côtés et pour chaque côté le site vert ou bleu peut être phosphorylé ou non. La réduction du modèle a donc remplacé une multiplication par une somme, ce qui aura un impact important lorsque le nombre de combinaisons possibles sera plus grand.

En guise de conslusion sur cet exemple, iIl est dont possible d'exploiter l'absence de flot d'information entre des sites des occurrences de protéines, afin de détecter des corrélations entre les états de plusieurs sites qui peuvent être ignorées. Cela revient à découper les configurations des espèces biochimiques en configuration de portions d'espèces biochimiques, et ainsi casser la combinatoire du système différentiel sous-jacent. Bien entendu, cette abstraction perd de l'information. Il n'est plus possible d'exprimer les corrélations qui ont été oubliées. Par contre, l'évolution de la quantité des configurations de portions d'espèces biochimiques qui auront été gardées se décrit de manière autonome, c'est à dire uniquement à partir de la quantité des configurations de portions d'espèces biochimiques elles-mêmes.

4.3 Réduction de systèmes d'équations différentielles

En section 4.3 est proposé un cadre formel pour réduire la dimension des systèmes d'équations différentielles ordinaires.

Les systèmes d'équations différentielles décrivent des quantités qui évoluent au cours du temps selon des contraintes sur la valeur de leurs dérivées. Une réduction de modèle consiste à trouver des changements de variables pour lesquels l'évolution de nouvelles quantités, appelées observables, peut se décrire de manière autonome. Ainsi la dérivée des observables doit pouvoir s'exprimer uniquement à partir de la valeur des observables.

Plus formellement, un système d'équations différentielles est donné par un ensemble fini de variables, $\mathcal{V}$, qui représentent ici la quantité de chacune des configurations des espèces biochimiques et par une fonction, $\mathbb{F}$, de l'ensemble des fonctions réelles positives ou nulles sur l'ensemble des variables $\mathcal{V}$ vers l'ensemble des fonctions réelles sur l'ensemble des variables $\mathcal{V}$. La fonction $\mathbb{F}$ est supposée dérivable et sa dérivée est continue. Une fonction réelle positive ou nulle sur l'ensemble $\mathcal{V}$ est appelée un état potentiel. En effet, un état associe à chaque variable, la quantité de l'espèce biochimique correspondante, qui ne peut pas être négative. Une fonction réelle sur l'ensemble $\mathcal{V}$ est appelée un incrément. Une telle fonction définit comment la quantité de chaque espèce

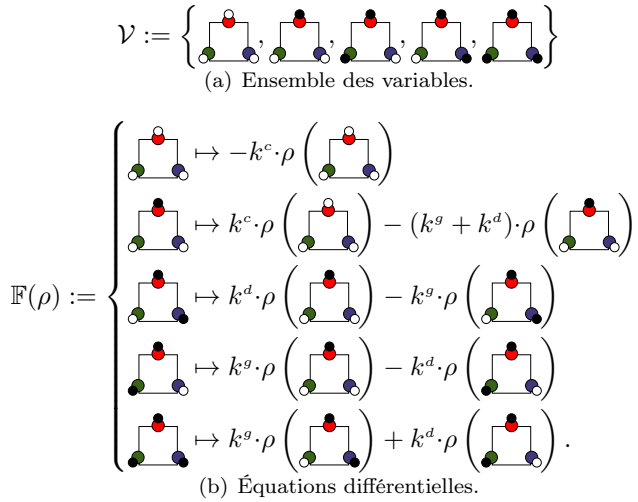


Figure 4.6: Système différentiel pour l'exemple jouet.

biochimique doit être augmentée ou diminuée sur un instant infinitésimal dt.

Exemple 4.3.1 *Dans notre exemple jouet, l'ensemble $\mathcal{V}$ et la fonction $\mathbb{F}$ sont donnés en figure 4.6(a). Il y a donc cinq variables, soit une variable pour la quantité de chaque configuration de la protéine. L'évolution de la valeur de ces variables est régie par les équations différentielles issues de l'application de la loi d'action de masse. Ces équations différentielles sont données en figure 4.6(b).*

La sémantique d'un système d'équations différentielles se définit comme la fonction qui, à un état initial X_0, associe la solution maximale, $X_{X_0}(T)$ de l'équation suivante :

$$X_{X_0}(T) = X_0 + \int_{t=0}^{T} \mathbb{F}(X_{X_0}(t))\cdot dt.$$

définie sur l'intervalle de temps $[0, T^{max}_{X_0}[$. Comme la fonction $\mathbb{F}$ est différentiable et de dérivée continue, cette équation définit un problème de Cauchy-Lipschitz [99]. Elle a donc une unique solution maximale. Ici l'adjectif 'maximal' signifie que la valeur du paramètre $T^{max}_{X_0}$ doit être prise la plus grande possible dans l'ensemble $\mathbb{R}^+ \cup \{+\infty\}$. Il y a donc deux possibilités. Soit la solution de cette équation contient une asymptote verticale auquel cas $T^{max}_{X_0}$ est la première date à laquelle la valeur d'une variable de l'ensemble $\mathcal{V}$ diverge. Soit la solution est définie sur tout $\mathbb{R}^+$ et dans ce cas $T^{max}_{X_0}$ vaut $+\infty$.

Exemple 4.3.2 *L'exemple introduit en figure 3.1(a) est maintenant utilisé pour illustrer la construction de la sémantique de différentielles. Il comporte trois espèces biochimiques A, B et C et les deux réactions suivantes :*

$$\begin{cases} A \to B \ \ @\, k_B \\ 2B \to C \ \ @\, k_C \end{cases}$$

La sémantique différentielle de ce réseau réactionnel est alors définie comme la solution du système différentiel suivant :

$$\begin{aligned} \frac{d\,[A]}{dt} &= -k_B \cdot [A] \\ \frac{d\,[B]}{dt} &= k_B \cdot [A] - 2{\cdot}k_C{\cdot}\,[B]^2 \\ \frac{d\,[C]}{dt} &= k_C \cdot [B]^2\,. \end{aligned}$$

Le comportement des systèmes réactionnels dont toutes les réactions de création (c'est à dire qui contiennent plus de produits que de réactifs) sont d'arité 0 ou 1 (c'est à dire sans réactif ou avec un seul réactif avec le

coefficient stœchiométrique 1), ne diverge pas. De ce fait, leurs sémantiques sont définies sur $\mathbb{R}^+$ quel que soient leurs états initiaux. En particulier, seul le comportement des systèmes ouverts (avec introduction externe de composants) est susceptible de diverger. Un exemple minimal de système dont le comportement diverge est donné en Exe. 4.3.3.

Exemple 4.3.3 *Soit le réseau réactionnel défini par l'unique réaction suivante :*

$$A, A \xrightarrow{k} A, A, A$$

portant sur une seule sorte de protéines, A.

La quantité de la protéine A est régie par l'équation différentielle suivante :

$$\frac{d\,[A]}{dt} = k \cdot [A]^2\,,$$

dont les solutions sont de la forme suivante :

$$[A] = \frac{[A]_0}{k - [A]_0 \cdot t}.$$

Chaque solution maximale diverge donc en $t = \dfrac{k}{[A]_0}$.

Réduire un système d'équations différentielles ordinaires consiste à changer de perspective en trouvant un ensemble de quantités d'intérêt, appelées les observables, dont l'évolution peut s'exprimer de manière autonome. Cela signifie que la dérivée des observables est entièrement définie par leurs valeurs. Pour formaliser ces notions, il faut d'une part, relier la valeur des observables aux variables du système d'équations différentielles initial et d'autre part, définir la fonction qui décrit l'évolution temporelle de ces valeurs.

Formellement, une réduction de modèle est définie par la donnée d'un ensemble fini d'*observables*, $\mathcal{V}^\sharp$, une fonction d'abstraction, ψ, qui associe à chaque état du système initial un état du nouveau système (c'est à dire une fonction de l'ensemble des fonctions positives ou nulles sur l'ensemble des observables $\mathcal{V}^\sharp$), et d'une fonction $\mathbb{F}^\sharp$ de l'ensemble des fonctions positives ou nulles sur l'ensemble des observables $\mathcal{V}^\sharp$ vers l'ensemble des fonctions réelles sur l'ensemble de observables $\mathcal{V}^\sharp$.

Des hypothèses supplémentaires sont requises pour garantir la correction de la réduction de modèle.

1. La fonction d'abstraction ψ doit être choisie linéaire à coefficients positifs. Elle doit de plus préserver les suites qui divergent, ce qui signifie que pour toute suite divergente de fonctions positives ou nulles $(\rho_i)_{i\in\mathbb{N}}$ sur l'ensemble des variables initiales $\mathcal{V}$, la suite $(\psi(\rho_i))_{i\in\mathbb{N}}$ diverge également.

2. Les fonctions d'abstraction ψ, la fonction de dynamique concrète $\mathbb{F}$ et la fonction de dynamique abstraite $\mathbb{F}^\sharp$ sont reliées par le diagramme commutatif suivant :

$$\begin{array}{ccc} (\mathcal{V} \to \mathbb{R}^+) & \xrightarrow{\mathbb{F}} & (\mathcal{V} \to \mathbb{R}) \\ \psi\downarrow & & \downarrow\psi \\ (\mathcal{V}^\sharp \to \mathbb{R}^+) & \xrightarrow{\mathbb{F}^\sharp} & (\mathcal{V}^\sharp \to \mathbb{R}) \end{array}$$

   ce qui signifie que $\psi \circ \mathbb{F} = \mathbb{F}^\sharp \circ \psi$.

Prendre la fonction d'abstraction ψ linéaire permet de la faire commuter avec la somme et l'intégrale utilisées pour définir la sémantique des systèmes différentiels. La prendre à coefficients positifs assure que l'image d'une fonction positive ou nulle est une fonction positive ou nulle, et donc que l'image d'un état concret sera un état abstrait. L'intérêt de l'hypothèse de préservation de la divergence des suites d'états sera expliqué plus tard. On peut toutefois remarquer que cette hypothèse revient à supposer que toute variable concrète apparait au moins une fois avec un coefficient non nul dans la combinaison linéaire associée à au moins un des observables. Enfin le diagramme commutatif permet à la fonction d'abstraction et à la fonction de dynamique concrète de commuter, au prix de transformer la dynamique concrète — sur les variables du système initiale — en dynamique abstraite — sur les observables. Dit autrement, partant d'un état concret, le même résultat est obtenu en calculant la dérivée du système dans cet état, puis en calculant l'abstraction de cette dérivée ; ou en calculant l'abstraction de l'état d'abord, puis en calculant la dérivée du système réduit dans l'état abstrait obtenu.

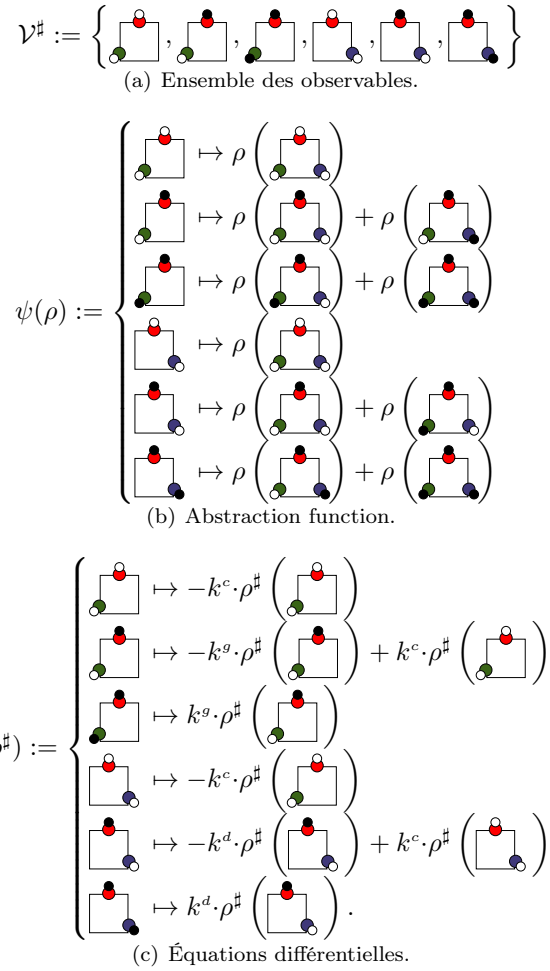


Figure 4.7: Reduction de la sémantique différentielle pour l'exemple jouet.

Exemple 4.3.4 *En figure 4.7 est donnée une réduction pour le système différentiel défini en figure 4.4. Les observables sont formés des motifs obtenus en prenant le site rouge, en haut, et en oubliant l'état du site vert, à gauche, ou l'état du site bleu, à droite. Ils sont donnés en figure 4.7(a).*

La quantité des observables est définie en fonction de la quantité des différentes configurations de la protéine en figure 4.7(b). En particulier, la quantité des motifs avec le site rouge non phosphorylé est définie comme la quantité de la configuration de la protéine dont aucun site n'est phosphorylé. Pour les autres motifs, cette quantité est définie comme la somme de la quantité des deux configurations qui contiennent ce motif. Il s'agit bien d'une fonction linéaire à coefficients positifs. Il faut remarquer que la concentration de chaque configuration d'espèces biochimiques apparaît au moins une fois dans la fonction ψ ce qui assure que cette fonction préserve la divergence des suites d'états.

Enfin, la dérivée des observables est donnée en figure 4.7(c). On peut vérifier par le calcul que $\psi \circ \mathbb{F} = \mathbb{F}^\sharp \circ \psi$, ce qui assure que le diagramme commute.

Il est maintenant possible d'appliquer la fonction d'abstraction ψ à gauche et à droite de l'égalité de Cauchy-Lipschitz qui avait servi à définir la sémantique du système initial.

Ainsi, pour X_0 un état initial (une fonction positive ou nulle sur l'ensemble $\mathcal{V}$) et pour T un instant de l'intervalle $[0, T_{X_0}^{max}[$, l'équation suivante est vérifiée :

$$\psi(X_{X_0}(T)) = \psi\left(X_0 + \int_{t=0}^{T} \mathbb{F}(X_{X_0}(t))\cdot dt\right).$$

Par linéarité de la fonction d'abstraction ψ, elle se distribue sur l'addition, ce qui donne l'équation suivante :

$$\psi(X_{X_0}(T)) = \psi(X_0) + \psi\left(\int_{t=0}^{T} \mathbb{F}(X_{X_0}(t))\cdot dt\right).$$

Toujour par linéarité, elle commute avec l'intégrale, ce qui donne l'équation suivante :

$$\psi(X_{X_0}(T)) = \psi(X_0) + \int_{t=0}^{T} \psi(\mathbb{F}(X_{X_0}(t)))\cdot dt.$$

Enfin, en utilisant le fait que $[\psi \circ \mathbb{F}] = [\mathbb{F}^\sharp \circ \psi]$, il vient l'équation suivante :

$$\psi(X_{X_0}(T)) = \psi(X_0) + \int_{t=0}^{T} \mathbb{F}^\sharp(\psi(X_{X_0}(t)))\cdot dt.$$

De ce fait, la fonction qui, à tout instant T dans l'intervalle $[0, T_{X_0}^{max}[$, associe l'état abstrait $\psi(X_{X_0}(T))$ est solution de l'équation différentielle suivante :

$$Y_{Y_0}(T) = Y_0 + \int_{t=0}^{T} Y_{Y_0}(t)\cdot dt,$$

pour $Y_0 = \psi(X_0)$.

C'est en fait une solution maximale de cette équation. En effet, c'est clair lorsque $T = +\infty$. Dans le cas contraire, l'expression $X_{X_0}(T)$ diverge quand T tend vers $T_{X_0}^{max}$. Puis comme la fonction ψ préserve la divergence des suites et qu'elle est continue, l'expression $\psi(X_{X_0}(T))$ diverge aussi quand T tend vers $T_{X_0}^{max}$. Ainsi la fonction qui à tout instant T dans l'intervalle $[0, T_{X_0}^{max}[$ associe l'état abstrait $\psi(X_{X_0}(T))$ ne peut pas être prolongée.

Exemple 4.3.5 *Pour finir cette partie, voici un exemple qui explique l'importance de l'hypothèse sur la préservation de la divergence des suites d'états.*

Soit le réseau réactionnel défini par les deux réactions suivantes :

$$\begin{array}{rcl} A, A & \stackrel{k_A}{\rightarrow} & A, A, A \\ B & \stackrel{k_B}{\rightarrow} & B, B \end{array}$$

portant sur deux sortes de protéines A et B.

La quantité de ces protéines est régie par l'équation différentielle suivante :

$$\begin{cases} \dfrac{d\,[A]}{dt} = k_A \cdot [A]^2 \\ \dfrac{d\,[B]}{dt} = k_B \cdot [B] \end{cases}$$

dont la solution maximale est donnée ci-dessous :

$$\begin{cases} [A] = \frac{[A]_0}{k - [A]_0 \cdot t} \\ [B] = [B]_0 \cdot e^k t \end{cases}$$

pour t variant dans l'intervalle de temps $\left[0, \frac{k}{[A]_0}\right[$.

En ignorant la contrainte sur la préservation de la divergence des suites d'états, il est possible d'oublier la quantité de la protéine A. Il suffit pour cela de prendre le singleton $\{B\}$ comme ensemble des observables et la fonction qui à chaque état du système associe sa restriction à l'ensemble $\{B\}$ comme fonction d'abstraction.

Le système réduit suivant est alors obtenu :

$$\frac{d\,[B]}{dt} = k_B \cdot [B].$$

Sa solution maximale est de la forme :

$$[B] = [B]_0 \cdot e^k t,$$

pour tout t dans l'ensemble $\mathbb{R}^+$.

Ce n'est dont pas l'image point à point de la solution maximale du système initial, par la fonction d'abstraction, mais un prolongement strict de cette dernière.

La sémantique d'un système différentiel et la réduction de celle-ci ont été formalisées. Il s'agit maintenant de définir la sémantique différentielle d'un système Kappa et de montrer comment trouver un ensemble d'observables dont l'évolution temporelle peut se décrire de manière autonome.

4.4 Application à Kappa

Il reste maintenant à spécialiser ce cadre générique pour réduire la sémantique différentielle des systèmes de règles du langage Kappa.

4.4.1 Sémantique différentielle

Comme il a été vu en exemple 2.9.1, un ensemble de règles Kappa engendre un réseau de réactions biochimiques. Chaque réaction étant alors constituée de deux n-uplets, l'un formé d'espèces appelées les réactifs de la réaction et l'autre d'espèces appelées les produits de la réaction. Pour définir la sémantique différentielle d'un ensemble de règles, il faut de plus associer à chacune de ces règles un constante d'interaction. Chaque réaction hérite alors de la constante de sa règle originale (ou de la combinaison linéaire des constantes de ses règles initiales si elle peut être obtenue de différentes manières).

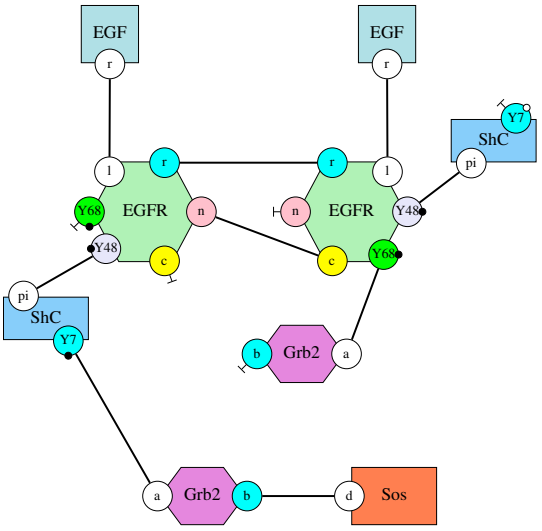
Comme expliqué en section 4.2, le comportement d'un réseau réactionnel est défini par l'application de la loi d'action de masse. La contribution de chaque réaction au système d'équations différentielles est définie de la manière suivante. Chaque réactif est alors consommé proportionnellement à l'activité de la réaction correspondante alors que chaque produit est ajouté selon la même quantité, l'activité de la règle s'exprimant comme le produit de la constante de la règle et de la quantité des réactifs (qui apparaissent dans le membre gauche de la règle).

Il existe plusieurs conventions pour tenir compte des éventuelles symétries dans les règles de réécriture et dans les réactions. Des précisions sur ce sujet peuvent être trouvées dans cette publication [15]. Nous négligeons ici la question en considérant que les taux des règles d'interactions et des réactions sont prises tels quels sans facteur correctif. La règle de trois permet de tenir compte des facteurs correctifs pour les autres conventions. Par ailleurs, toutes les réactions ne sont pas prises en compte. En effet, seules les réactions qui ont le même nombre de réactifs que le nombre de composante connexe dans le membre gauche de la règle qui les a engendrées, sont prises en compte. Intuitivement, la sémantique différentielle représente le comportement du système discret dans un milieu homogène et parfaitement fluide dont le volume tend vers 0. Aussi, la chance de tirer au sort deux parties d'une même occurrence de configuration d'espèces biochimique pour plonger plusieurs composantes connexes du membre gauche d'une règle d'interaction tend vers 0.

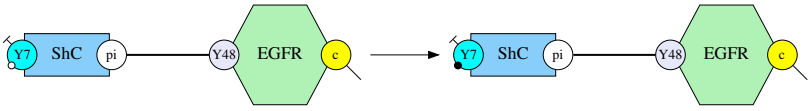
4.4.2 Inférence du flot d'information

En Kappa, une approximation supérieure du flot d'information entre les sites des configurations d'espèces biochimiques peut être obtenue en inspectant chaque règle d'interactions. En effet, il ne peut y avoir un flot d'information d'un premier site vers un second site dans la configuration d'une espèce biochimique que s'il existe existe un règle qui transforme l'état du second site selon des conditions sur l'état du premier site.

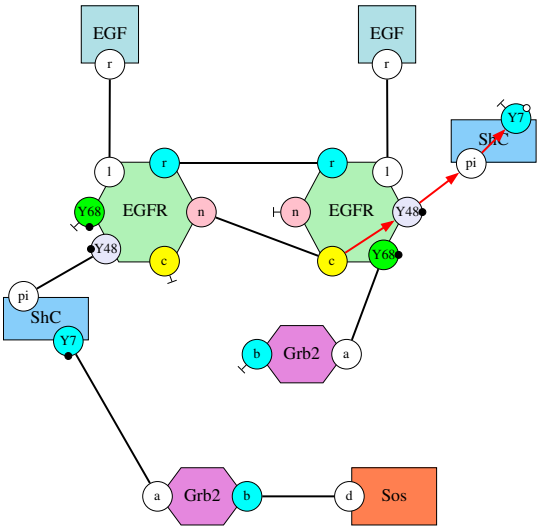
Exemple 4.4.1 *Par exemple, dans la configuration suivante de dimère :*



La règle suivante :



ne peut phosphryler le site Y7 de l'occurrence de la protéine ShC que si le site c l'occurrence de droite de la protéine récepteur EGFR est lié. Ceci justifie le flot d'information du site c de l'occurrence de droite de la protéine récepteur vers le site Y7 de l'occurrence de protéine ShC. Ce flot d'information prend la forme d'un chemin dans l'annotation du graphe à sites, comme représenté ci-dessous :



De ce fait, il est possible d'annoter la configuration d'une espèce biochimique par une approximation supérieure de son flot d'information en calculant toutes les instances des composantes connexes du membre gauche de chaque règle d'interactions du modèle et en reportant le long des plongements correspondants chaque chemin constitué d'étapes entre deux sites d'une même occurence de protéines ou entre deux sites sur un même lien et qui se termine dans un site dont l'état est modifié par la règle. Ainsi, si deux sites de la configuration de l'espèce biochimique sont l'image par un plongement de deux sites reliés par une étape de ce chemin, un arc de flot d'information est placé entre ces deux sites dans la configuration de l'espèce biochimique. Cette construction est illustrée en figure 4.8.

Il n'est cependant pas concevable d'annoter ainsi toutes les configurations d'espèces biochimiques, car celles-ci sont trop nombreuses en général. Une telle approche ne passerait pas à l'échelle de modèles d'interactions biochimiques même de taille modeste. Il est possible au contraire de résumer l'annotation de toutes les configurations d'espèces biochimiques sur la carte de contacts. Il faut se souvenir que la carte de contacts fournit

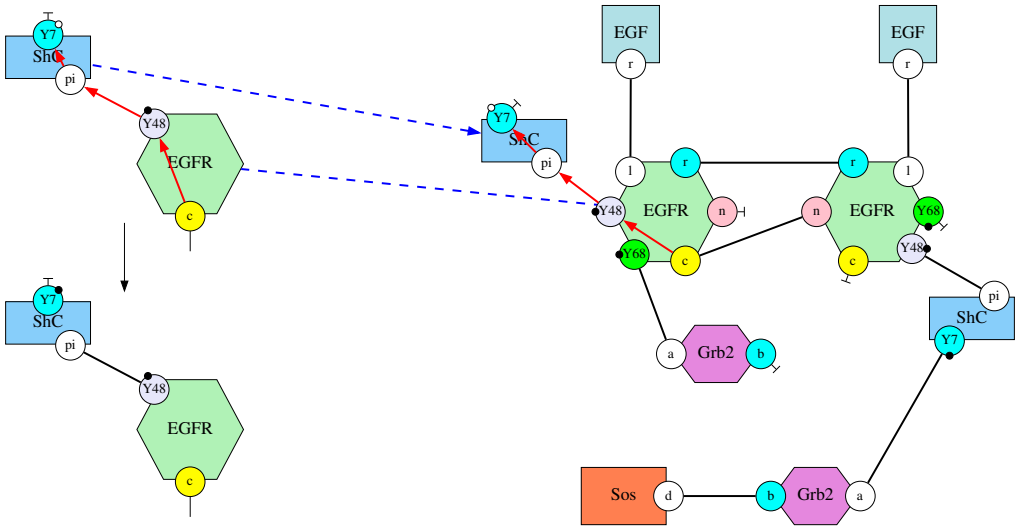


Figure 4.8: Flot d'information induit par une règle d'interactions sur la configuration d'une espèce biochimique. Pour tout plongement d'une composante connexe du membre gauche de la règle (ici dessiné en haut) vers la configuration de l'espèce biochimique et tout chemin allant d'un site, ici le site c de l'occurrence du récepteur $EGFR$, et un site dont l'état est modifié par la règle, ici le site $Y7$ de l'occurrence de la protéine d'échafaudage ShC, le chemin est reporté sur la configuration de l'espèce biochimique selon le plongement.

une abstraction de l'ensemble de toutes les configurations des espèces biochimiques d'un modèle en repliant entre eux les occurrences de chaque sorte de protéines. La carte de contacts peut être calculée directement par inspection des états initiaux et des règles du modèle, ou par analyse statique, quitte à introduire des états fictifs. Comme tout motif se projète de manière unique sur la carte de contacts en envoyant toutes les occurrences d'une même protéine sur l'unique occurrence de cette protéine dans le carte de contacts, cette projection permet de répertorier le flot d'information induit par une règle directement sur la carte de contacts. Ainsi tout arc qui apparaît dans un chemin constitué d'étapes entre deux sites d'une même occurence de protéines ou entre deux sites sur un même lien, partant d'un site dans un composante connexe dans le membre gauche d'une règle et finissant sur un site dont l'état est modifié par la règle de réécriture, est reporté sur la carte de contacts comme un flot d'information potentiel entre l'image de ces deux sites. Cette construction est illustrée en figure 4.9.

Le cas des composantes connexes qui ne sont pas modifiées dans une règle est particulier. Il n'y a en effet aucun chemin de flot d'information dans celles-ci. Pour garantir qu'il sera possible d'exprimer la quantité des ces motifs dans le système réduit, il faut considérer qu'au moins un site d'interraction, au choix, pour chacune de ces composantes connexes est modifié, et donc reporter tous les chemins partant des autres sites et terminant sur ce site dans la carte de contacts annotée.

Ainsi, il est possible de résumer de le flot d'information induit par les règles entre les différents sites des configurations des espèces biochimiques sur la carte de contacts. Il n'est pas nécessaire d'énumérer les différentes configurations des espèces biochimiques. Il suffit en effet d'identifier les chemins dont chaque étape relie soit deux sites d'une même occurrence de protéines, soit deux sites qui partagent un lien, et qui se terminent sur un site qui est modifié par la règle, et de répertorier ce chemin sur la carte de contacts.

4.4.3 Pré-fragments et fragments

Réciproquement, l'approximation supérieure du flot d'information sur la carte de contacts permet de reconstruire une approximation supérieure du flot d'information sur n'importe quel motif. Il suffit pour cela de considérer l'image inverse de la projection qui envoie ce motif sur la carte de contacts. Ainsi, il y aura un arc de flot d'information entre deux sites d'un motif si et seulement si il y a un arc de flot d'information sur l'image de ces deux sites par sa projection sur la carte de contacts. Cette construction est illustrée en figure 4.10.

L'annotation d'un motif par son flot d'information permet de vérifier si d'une part, il ne comporte pas de sites dont la corrélation entre les états ne présente pas d'intérêt vis à vis du comportement du système modélisé, et si d'autre part, il contient tous les sites d'interactions susceptibles d'influencer son comportement. Un motif sera gardé comme observable lorsque ces conditions seront toutes deux réalisées. Formellement, la première

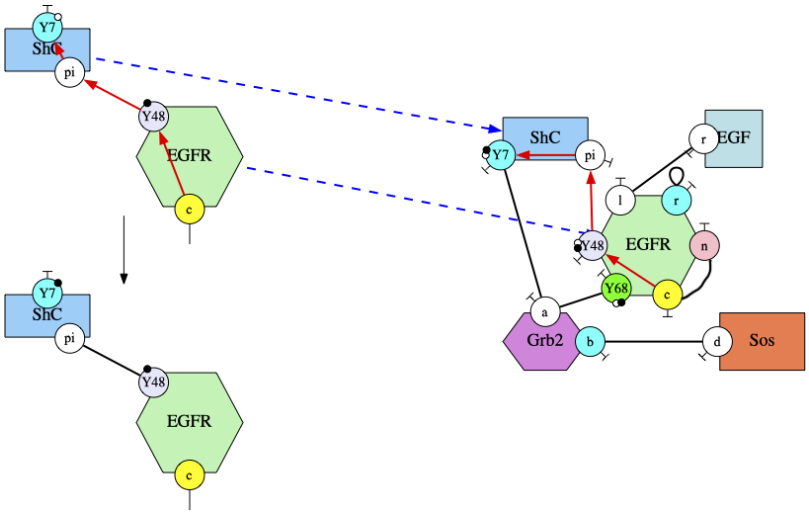


Figure 4.9: Flot d'information induit par une règle sur la carte de contacts. Pour tout chemin allant d'un site, ici le site c de l'occurrence du récepteur $EGFR$, et un site dont l'état est modifié par la règle, ici le site $Y7$ de l'occurrence de la protéine d'échafaudage ShC, le chemin est reporté sur la carte de contacts en suivant l'unique projection du membre gauche de la règle sur la carte de contacts.

condition est satisfaite si le graphe formé par l'annotation du motif n'admet qu'une seule composante fortement connexe terminale. Dans ce cas, le motif sera appelé un *pré-fragment*. La seconde condition est réalisée lorsqu'un pré-fragment ne se plonge pas dans un autre pré-fragment. C'est à dire, s'il est impossible de compléter le motif, sans briser la condition que le graphe formé par l'annotation de ce motif par son flot d'information n'admet qu'une seule composante fortement connexe terminale. Dans ce cas, le motif sera appelé un *fragment*.

Exemple 4.4.2 *En figure 4.11 sont dessinés quatre motifs. Le but est de comprendre parmi ceux-ci, lesquels sont des fragments. Pour répondre à cet question, il faut les annoter par le flot d'information répertorié sur la carte de contacts. Seuls les motifs pour lesquels le graphe formé par l'annotation du flot d'information n'a qu'une seule composante fortement connexe terminale sont des pré-fragments.*

Les motifs annotés avec le flot d'information ainsi obtenu sont représentés en figure 4.12, ainsi que les compostantes fortement connexes du graphe de leur flot d'information. Il apparaît que celui du troisième motif a deux composantes fortement connexes terminales. Ce n'est donc pas un pré-fragment. En revanche, les trois autres motifs le sont bien.

Le deuxième motif se plonge dans le premier motif. Ce n'est dont pas un fragment. Par contre, le premier et le quatrième motif sont des fragments. En effet, si on leur ajoute une occurrence de site, ce sera forcément soit une occurrence du site Y48 soit une occurrence du site Y68. Dans les deux cas, cela ajoutera une composante fortement connexe terminal dans le graphe du flot d'information du motif ainsi obtenu. De ce fait, ces motifs ne se plongent pas dans des fragments, autres qu'eux-même.

Ainsi seuls les deux motifs représentés dans les figures 4.11(a) et 4.11(d) sont des fragments.

4.4.4 Sémantique différentielle réduite

Il reste à montrer que l'ensemble des fragments d'un modèle peut être pris comme ensemble des observables de la sémantique différentielle réduite. En d'autres termes, il faut montrer qu'il est possible d'exprimer la quantité de chaque fragment crée et détruite sur un temps infinitésimal en fonction de la quantité des autres fragments dans l'état du système.

4.4.4.1 Raffinements orthogonaux

Avant tout, il est important de remarquer que la quantité de chaque pré-fragment s'exprime comme une combinaison linéaire de la quantité des fragments. Cette propriété repose sur l'utilisation d'arbres de décision pour raffiner étape par étape un pré-fragment en un ensemble de pré-fragments orthogonaux plus précis [52, 112, 79]. Par définition, un pré-fragment qui n'est pas un fragment se plonge dans un pré-fragment plus grand. Ce

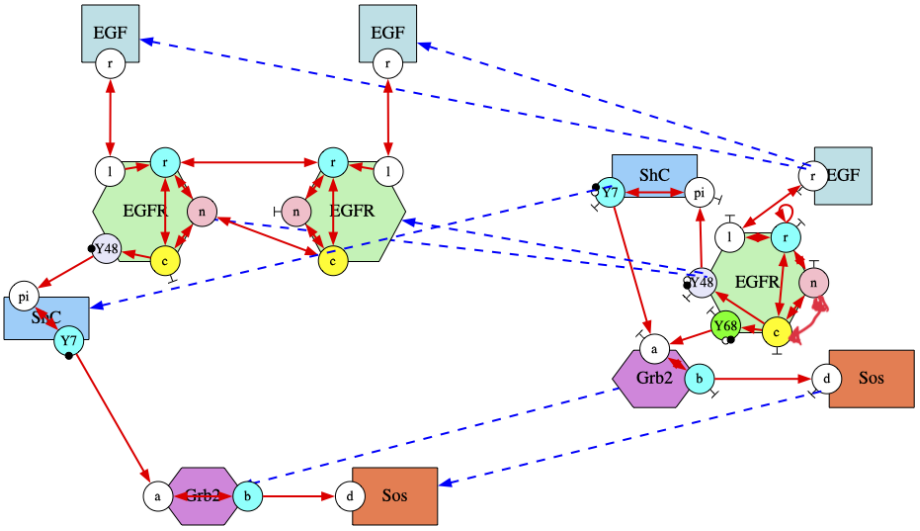
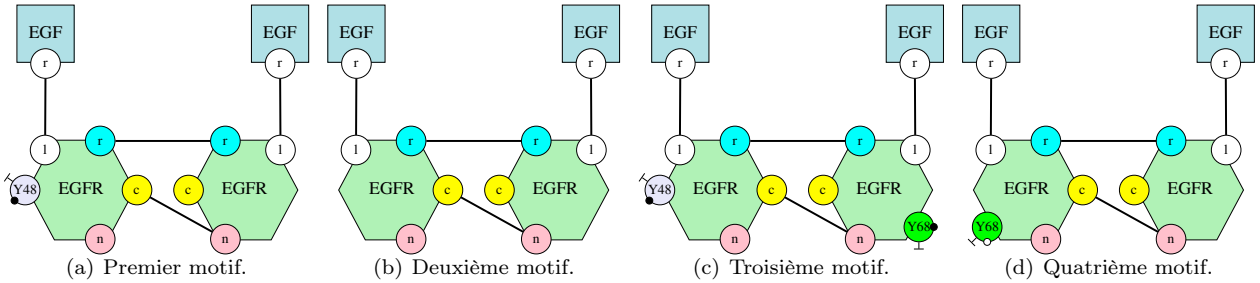


Figure 4.10: Annotation d'un motif par le flot d'information décrit dans la carte de contacts. Le motif se plonge de manière unique dans la carte de contacts en envoyant chaque occurrences d'agent sur l'unique occurrence de cet agent dans le carte de contacts. Deux sites du motifs sont reliés par un arc de flot d'information si et seulement si ils sont sur le même agent ou sur une même liaison et si leur image dans la carte de contacts est lié par un arc de flot d'information.



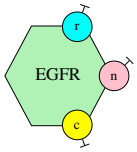
(a) Premier motif. (b) Deuxième motif. (c) Troisième motif. (d) Quatrième motif.

Figure 4.11: Parmi ces quatre motifs, lesquels sont des fragments ?

dernier contient donc un site qui n'est pas présent dans le pré-fragment initial. Le pré-fragment initial peut alors être remplacé par l'ensemble des pré-fragments obtenus en ajoutant ce site pour tous les états possibles de ce sites. L'arbre de décision est ainsi construit par induction jusqu'à ce que les feuilles de l'arbre soient toutes des fragments. Pour chaque nœud, la quantité du motif est égale à la somme des quantités des motifs sur ses feuilles, ce qui garantit que la quantité du pré-fragment initial est égal à la somme des quantités des fragments sur les feuilles de l'arbre. À noter que certains fragments peuvent apparaître plusieurs fois dans les feuilles de l'arbre, ce qui peut donc donner des coefficients différents de 1 dans la combinaison linéaire ainsi obtenue.

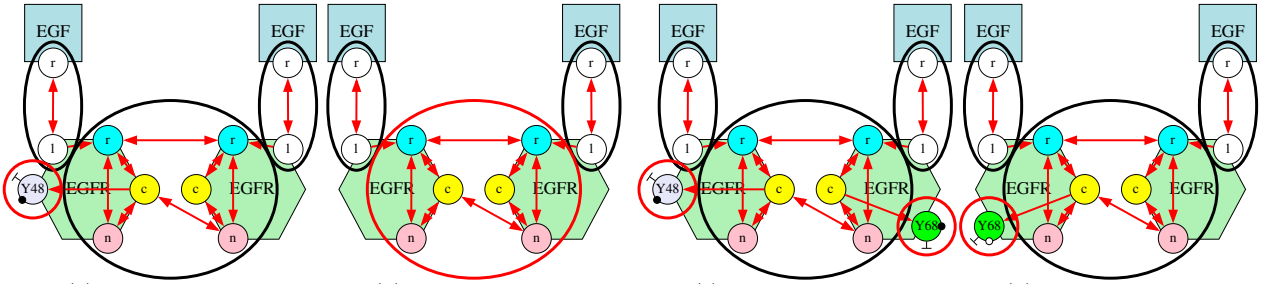
Exemple 4.4.3 *En figure 4.13 est donné un exemple d'arbre de décision pour exprimer la quantité d'un pré-fragment comme une combinaison linéaire de la quantité des fragments.*

La racine de l'arbre est le pré-fragment suivant :



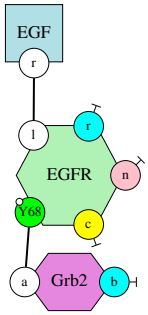
formé d'une occurrence de la protéine récepteur dont les sites r, c et n sont libres. L'état des autres sites n'est pas précisé.

Le fragment suivant :



(a) Premier motif. (b) Deuxième motif. (c) Troisième motif. (d) Quatrième motif.

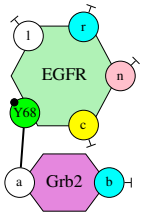
Figure 4.12: Pour décider lesquels de ce motifs sont des fragments, la première étape est de reporter l'annotation du flot d'information entre les sites d'interactions de la carte de contacts sur ces motifs et de décomposer le graphe obtenu en composantes fortement connexes.



dans lequel ce pré-fragment se plonge, sert de référence pour construire la partie centrale de l'arbre de décision. Dans ce fragment, le site l de l'occurrence de la protéine récepteur est lié au site r d'une occurrence de la protéine EGF et le site Y68 est phosphorylé et lié au site a d'une occurrence d'une protéine de transport dont le site b est libre.

La construction de la partie centrale de l'arbre de décision se fait de la manière suivante. Dans un premier temps est considéré l'état de phosphorylation du site Y68 de l'occurrence de la protéine EGFR. Ce site peut être phosphorylé ou non. Dans le cas où le site est phosphorylé, la seconde information considérée est l'état de liaison du site l. Celui-ci peut être libre ou lié au site r d'une occurrence de la protéine EGF. Lorsqu'il est lié, la troisième information considérée est le fait que le site Y68 soit libre ou lié au site a d'une occurrence de la protéine de transport. Enfin, dans ce cas, le site b de cette occurrence peut être libre ou lié au site d d'une occurrence de la protéine Sos.

La partie supérieure de l'arbre de décision correspond au cas où le site Y68 est phosphorylé et le site l est libre. Dans ce cas, la construction de l'arbre de décision peut être guidée par le fragment suivant :



Ce fragment est identique au précédent hormi le fait que le site l de l'occurrence de la protéine EGFR est libre. La construction de la partie supérieure de l'arbre de décision se fait alors de la même manière que la partie centrale.

Chaque sous-arbre peut être construit de manières différentes, soit en posant des questions différentes, soit en les posant dans un ordre différent. C'est le cas de la partie inférieure de l'arbre qui est construite à partir du fragment suivant :

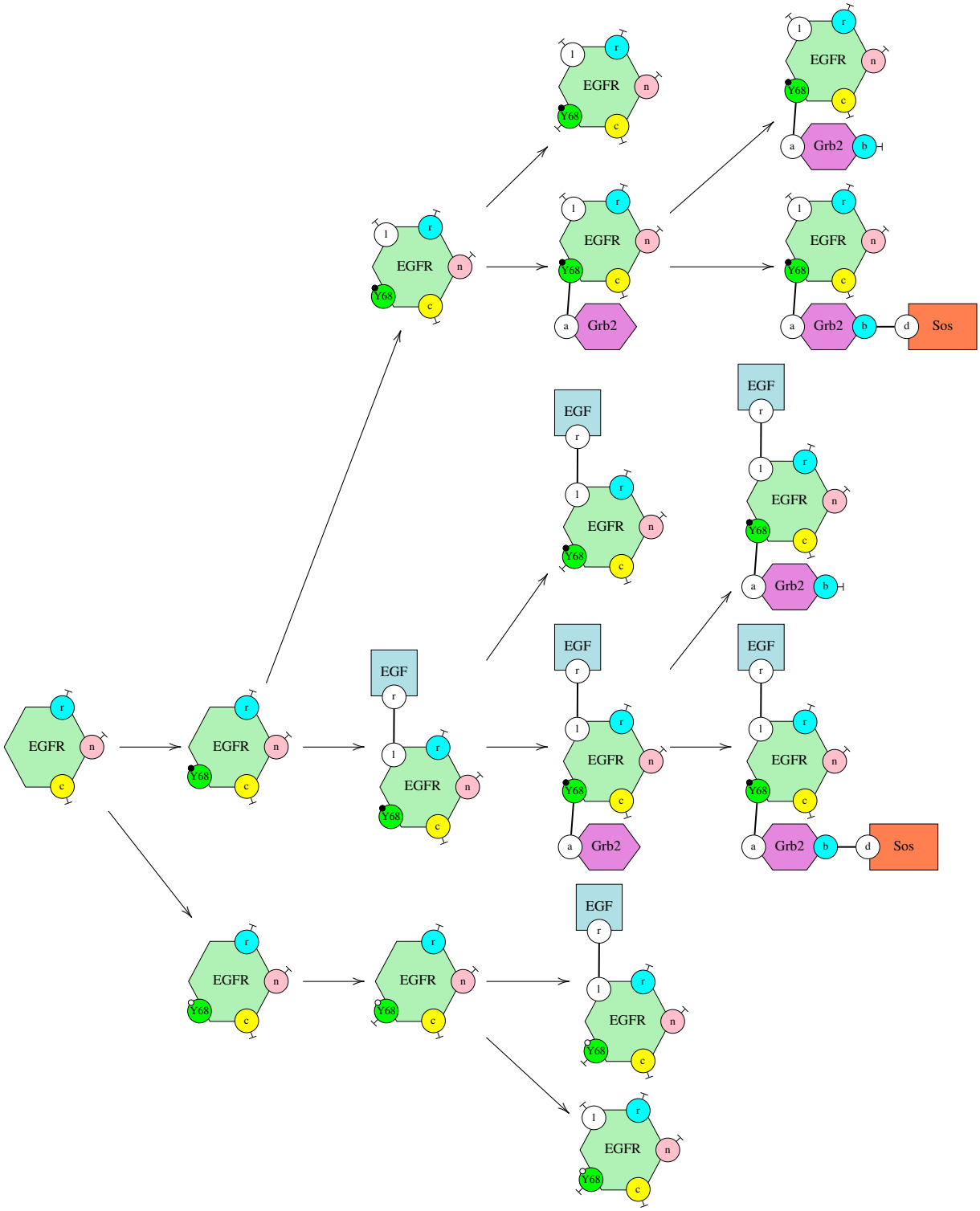
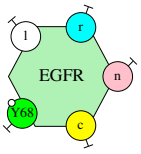


Figure 4.13: Un arbre de décision pour exprimer la quantité d'un pré-fragment, la racine de l'arbre, comme la combinaison linéaire de la quantité de fragments, les feuilles de l'arbre. L'arbre de décision est obtenu en discutant de l'état de phosphorylation du site *Y68* de l'occurrence de la protéine récepteur. La partie supérieure est ensuite obtenue en discutant de l'état du liaison du site l, puis de l'état de liaison du site *Y68*, puis quand ce site est lié de l'état de liaison du site b de l'occurrence de la protéine à laquelle ce site est lié. La partie inférieure est obtenue en discutant de l'état de liaison du site *Y68*, puis de l'état de liaison du site l.



Dans cette partie de l'arbre, l'état de liaison du site Y68 de l'occurrence de la protéine EGFR est discuté avant celui de son site l.

4.4.4.2 Spécialisation d'une règle à la consommation ou la production d'un pré-fragment

La seconde étape consiste à exprimer la consommation et la production de la quantité de chaque fragment sur un temps infinitésimal, en fonction de la quantité des autres fragments dans l'état du système. Cette partie présente un résultat plus général qui exprime la quantité consommée et produite pour chaque pré-fragment en fonction de la quantité des autres pré-fragments. Ceci répond bien à la question puisque la partie précédente a permis de traduire la quantité de chaque pré-fragment comme une combinaison linéaire de la quantité des fragments du modèle.

Cette approche repose sur une spécialisation des règles du modèle à la consommation ou à la production d'une occurrence d'un pré-fragment à une position donnée. En effet, chaque chevauchement potentiel entre un pré-fragment et le membre gauche ou droit d'une règle permet de raffiner celle-ci en ajoutant des deux côtés de cette règle toute information présente dans le pré-fragment qui ne serait pas dans le membre en question. Cette construction avait déjà été utilisée page 33 pour détecter parmi un ensemble de motifs d'intérêts certains qui ne sont pas accessibles dans un modèle.

Plus formellement, un chevauchement entre deux motifs se caractérise par un troisième motif et deux plongements des deux premiers motifs vers ce troisième motif, de sorte qu'aucune information qui ne serait présente ni dans le premier motif, ni dans le second ne soit présente dans le troisième. En d'autre terme, le troisième motif doit être minimal. Dans la théorie des catégories, ceci correspond à une *somme amalgamée*.

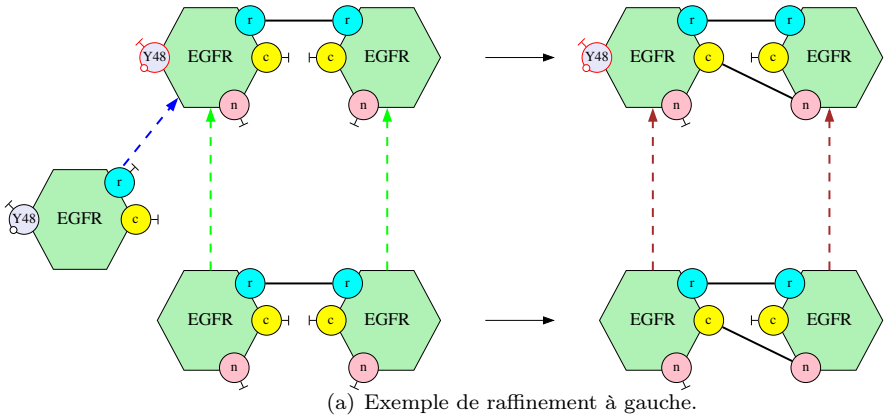
Exemple 4.4.4 *En figure 4.14 sont donnés des exemples de raffinements de règles.*

Le premier est une spécialisation de la règle de formation du lien asymétrique dans les dimères (voir en figure 2.8(e) page 20) à la consommation des occurrences de la protéine EGFR dont les sites d'interactions r et c sont libres et le site Y48 libre et non phorphorylé (l'état des sites l, n et Y68 ne sont pas mentionnés) en première position du membre gauche de la règle. Il faut pour cela considérer le chevauchement obtenu en fusionnant l'occurrence de la protéine EGFR du motif avec la première occurrence de cette protéine dans le membre gauche de la règle (un autre chevauchement pourrait être obtenu de la même manière en l'identifiant à la seconde occurrence de cette protéine). Les seules informations qui sont présentes dans le motif sans l'être dans le membre gauche de la règle sont que le site Y48 doit être libre et non phosphorylé. Le raffinement de la règle est donc obtenu en ajoutant ces informations à gauche et à droite de la règle. Le résultat est donc une spécialisation de la règle de liaison asymétrique au cas où la première occurrence de la protéine EGFR a son site Y68 libre et phosphorylé.

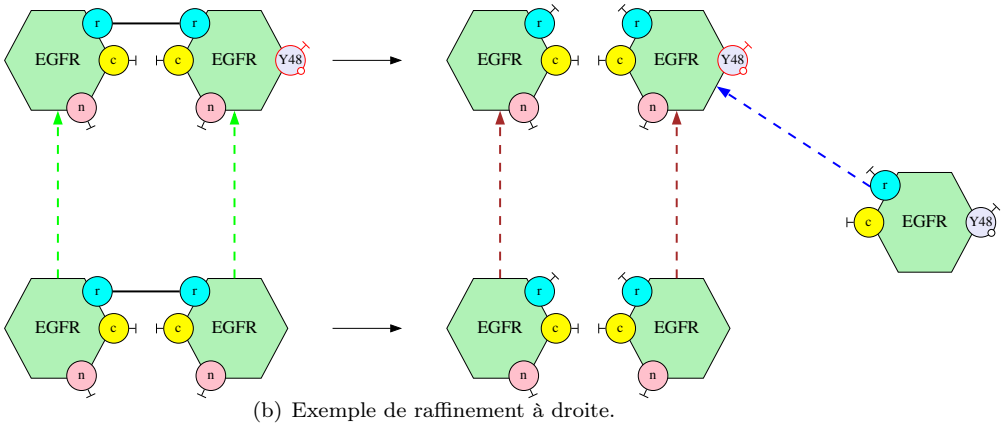
Le second raffinement est une spécialisation de la règle pour briser un lien symétrique dans les dimères (voir en figure 2.8(d) page 20) à la production des occurrences du même motif en deuxième position du membre droit de la règle. Il résulte du chevauchement obtenu en fusionnant l'occurrence de la protéine EGFR du motif avec la seconde occurrence de cette protéine dans le membre droit de la règle. Les seules informations qui sont présentes dans le motif sans l'être dans le membre gauche de la règle sont que le site Y48 doit être libre et non phosphorylé. Le raffinement de la règle est donc obtenu en ajoutant ces informations à gauche et à droite de la règle. Le résultat est donc une spécialisation de la règle pour briser la liaison symétrique dans les occurrences de dimères dans le cas où la deuxième occurrence de la protéine EGFR a son site Y68 libre et phosphorylé.

4.4.4.3 Termes de consommation et de production d'un motif

Les raffinements d'une règle par un motif permettent d'exprimer la quantité de ce motif consommée et la quantité produite sur un temps infinitésimal par cette règle en fonction de la quantité des autres motifs dans l'état du système. De plus lorsque ce motif est un pré-fragment et que le chevauchement entre le pré-fragment et le membre de la règle, qui a induit le raffinement, contient un site modifié par la règle sur sa partie commune, alors par construction du flot d'information, les composantes connexes de la règle raffinée sont toutes des pré-fragments. Comme la quantité de chaque pré-fragment s'exprime comme un combinaison linéaire de la quantité



(a) Exemple de raffinement à gauche.



(b) Exemple de raffinement à droite.

Figure 4.14: Deux raffinements de règles d'interactions. En 4.14(a), la règle de formation de liaison asymétrique dans les occurrences de dimères (voir la figure 2.8(e) page 20) est raffinée en faisant se chevaucher un motif avec son membre gauche. En 4.14(b), la règle pour briser la liaison symétrique dans les occurrences de dimères (voir la figure 2.8(d) page 20) est raffinée en faisant, cette fois ci se chevaucher ce motif avec le membre droit de la règle. Dans les deux cas, la partie commune au motif et au membre de la règle contient un site dont l'état est modifié par la règle d'interaction. De plus, l'information présente dans le motif qui manque dans le membre de la règle est ajoutée (en rouge) aux deux membres de la règle pour spécialiser celle-ci à la consommation ou à la production de ce motif par la règle.

des fragments, ceci permet d'exprimer la consommation et la production de chaque fragment sur un temps infinitésimal en fonction de la quantité des autres fragments. Par ailleurs, comme la sémantique différentielle ne prends en compte que les régles-réactions qui ont le même nombre de composantes connexes dans leur membre gauche que la règle dont elles sont issues, seuls les raffinements qui préservent le nombre de composantes connexes dans le membre gauche ont une contribution dans le système différentiel réduit.

Ainsi, pour chaque chevauchement entre un pré-fragment et le membre gauche d'une règle tels qu'un site modifié appartienne à la partie commune et que les membres gauches de la règle initiale et de la règle raffinée aient le même nombre de composantes connexes, si l'on note k la constante de la règle, aut le nombre d'automorphismes du pré-fragment et $C'_1,\ldots,C'_n$ les pré-fragments qui constituent le membre gauche de la règle raffinée, la consommation de ce pré-fragment sur un temps infinitésimal est donnée par l'expression suivante :

$$\frac{k \cdot \Pi_{i=1}^{n} [C_i]}{aut}.$$

De la même manière, pour chaque chevauchement entre un pré-fragment et le membre droit d'une règle tels qu'un site modifié appartienne à la partie commune et que les membres gauches de la règle initiale et de la règle raffinée aient le même nombre de composante connexe, si l'on note k la constante de la règle, aut le nombre d'automorphismes du pré-fragment et $C'_1,\ldots,C'_n$ les pré-fragments qui constituent le membre gauche de la règle

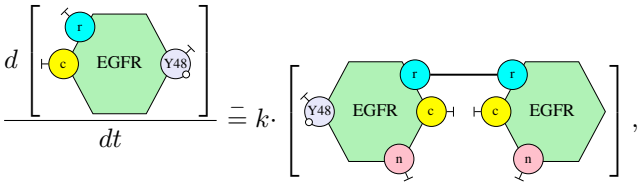
raffinée, la production de ce pré-fragment sur un temps infinitésimal est donnée par l'expression suivante :

$$\frac{k \cdot \Pi_{i=1}^{n} [C_i]}{aut}.$$

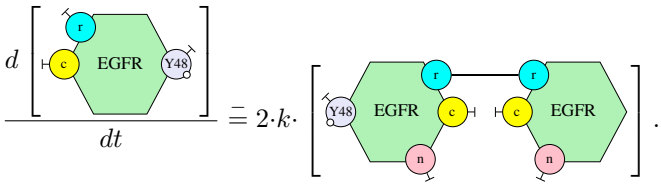
Ainsi le principe de la loi d'action de masse opère directement sur les quantité de motifs, ce qui permet d'exprimer directement un système d'équations différentielles ordinaires réduit. Appliqué à des fragments, il offre, par construction, une abstraction exacte du système d'équations différentielles initial. Il est important de noter que ce qui a été réalisé est en fait une factorisation de l'expression de la dérivée des quantités de fragments. En effet, cette dérivée peut s'exprimer en sommant l'activité de toutes les règles-réactions dans lesquels ce fragment est produit ou consommé. Chaque activité est alors le produit d'une constante et de quantités de configurations d'espèces biochimiques. En exprimant cette dérivée par des concentrations de fragments, des configurations d'espèces biochimiques ont été regroupées facteur par facteur pour retrouver des motifs, transformant ainsi des sommes de produits en produits de sommes. La préservation des termes de l'expression initiale est une preuve purement combinatoire qui constitue le chapitre 8.3 de [23].

Enfin, un fragment peut apparaître dans un réaction sans être consommé, ni produit. Il est en général impossible d'exprimer la quantité correspondante en fonction de la quantité des fragments. Cela n'a pas d'incidence, car cela conduit à des contributions négatives et positives qui s'annule parfaitement dans le système d'équations différentielles initial.

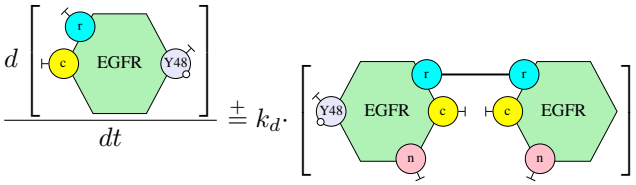
Exemple 4.4.5 *Pour continuer l'exemple 4.4.4, il est possible d'exprimer le terme de consommation et de production liés aux raffinements représentés en figure 4.14. Ainsi le raffinement gauche en figure 4.14(a) engendre le contribution suivante :*



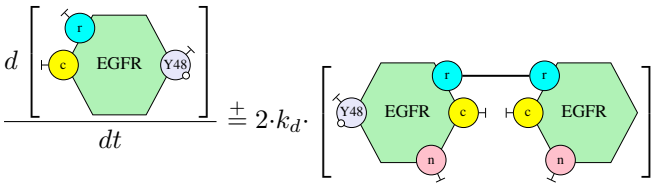
où k est la constante de la règle de formation de la liaison asymétrique dans les dimères. Par ailleurs, en tenant compte du chevauchement possible entre le même motif et la seconde occurrence de la protéine EGFR dans le membre gauche de la règle, on obtient la contribution globale suivante :



De même, le raffinement droit en figure 4.14(b) engendre le contribution suivante :



où k_d est la constante de la règle pour briser les liaisons symétriques dans les dimères. Par ailleurs, en tenant compte chevauchement possible entre le même motif et la première occurrence de la protéine EGFR dans le membre droit de la règle, on obtient la contribution globale suivante :



modèle	configurations d'espèces biochimiques	fragments	temps pour générer le système réduit
egfr (simplifié)	356	38	0.3 s.
egfr	1232	238	0.2 s.
egfr, erk, mapk, ras	$\sim 2.10^{19}$	$\sim 2.10^{5}$	180 s.

Table 4.1: Résultats expérimentaux. Pour chaque modèle est donné le nombre de configurations différentes d'espèces biochimiques, le nombre de fragments et le temps passé pour générer le système réduit d'équations différentielles sur un MacBook Pro avec une puce Intel Core i7-6567U (cadencée 3.3 GHz)).

4.5 Étude de performance

Cette approche a été implantée dans un outils disponible dans une ancienne version de Kappa. Nous comptons la porter dans la nouvelle version, mais ce n'est fait à l'heure de l'écriture de cet inédit.

En table 4.1, sont donnés des résultats expérimentaux pour la synthèse des systèmes réduits d'équations différentielles pour trois modèles. Pour chaque modèle sont donnés le nombre total de configurations d'espèces biochimiques, le nombre total de fragments, et le temps de calcul utilisé à générer le système différentiel réduit (sur un MacBook Pro avec une puce Intel Core i7-6567U (cadencée 3.3 GHz)).

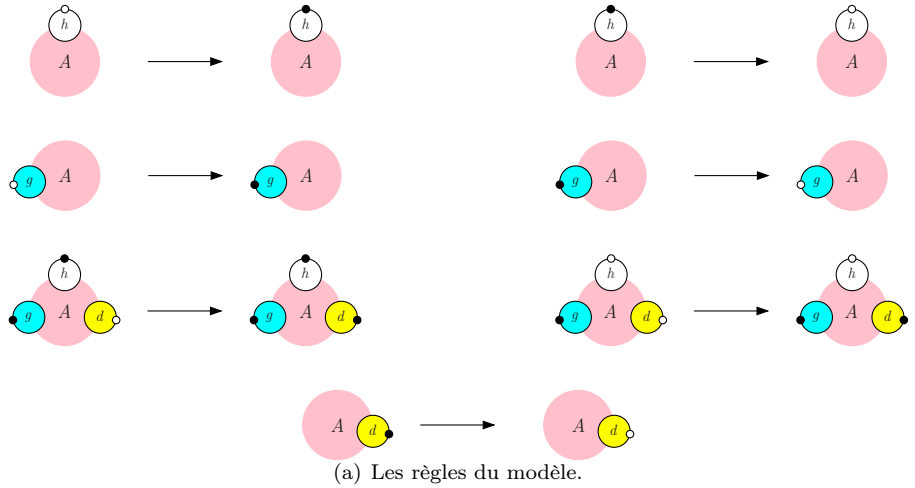
Les deux premiers modèles sont des versions de modèles pour les premières étapes de l'intégration du facteur de croissance de l'épiderme. Dans la première version, la formation des occurrences du dimère est simplifiée car elle ne comporte qu'une liaison symétrique. Dans la seconde version, à la fois les liaisons symétriques et asymétriques sont considérées. Enfin le troisième modèle est une version beaucoup plus complète de la voie de signalisation incluant plusieurs étapes après le recrutement des occurrences de la protéine *Sos* [51, 10, 124, 20].

4.6 Pour aller plus loin

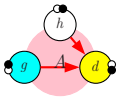
Dans cette partie, une méthode de réduction de la sémantique différentielle des modèles Kappa a été présentée. Elle se base sur une analyse du flot d'information entre les différents sites d'interactions des configurations des espèces biochimiques pour identifier des corrélations entre l'état de certains sites qui sont inutiles pour décrire le comportement temporel du modèle. Ceci permet d'identifier des motifs d'intérêt, appelés fragments, dont l'évolution peut être décrit uniquement en fonction de la quantité des fragments dans le système, et donc sans connaître la quantité de chaque configuration d'espèces biochimiques. Ainsi, il est possible de générer automatiquement un système réduit d'équations différentielles décrivant l'évolution de la quantité de ces fragments au cours du temps. Il suffit pour cela de spécialiser les règles à la consommation ou à la production des fragments aux différentes positions compatibles sur la règle. Ceci évite d'avoir à énumérer les réactions du modèle ou l'ensemble des configurations d'espèces biochimiques. Cette méthode a réussi à exploiter la structure en cascade du flot d'information pour réduire la dimension de modèles à large échelle de voies de signalisation intracellulaire.

La présentation de la méthode a été simplifiée en ignorant les règles avec effets de bords. Ces règles permettent de détruire des occurrences de protéines sans en décrire entièrement l'état ou de briser une liaison sur un site sans spécifier à quel site ce site est lié. Ceci pose deux difficultés techniques. Premièrement la spécialisation des règles pour la production d'un fragment à une certaine position peut donner lieu à plusieurs raffinements de la règle en fonction des effets de bord à effectuer. Ensuite, la description précise de ces effets de bord impose d'étendre la syntaxe pour quantifier existentiellement sur la présence de liens non spécifiés entre des agents. Toutefois, il a été montré dans [25] que la quantité de ces motifs étendus pouvait s'exprimer sous forme de somme alternée de motifs classiques grâce au développement de la formule du binôme.

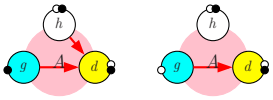
La méthode présentée se repose sur une approximation uniforme du flot d'information. En effet, le flot d'information est résumé sur la carte de contacts, or celle-ci ne comporte qu'une occurrence de chaque sorte des différentes protéines. Une réduction de modèle plus compacte peut parfois être obtenue en utilisant une approximation non uniforme du flot d'information [26, 23]. Le calcul des fragments est alors un peu plus compliqué. Il faut notamment appliquer un opérateur de clôture sur l'approximation du flot d'information afin de garantir que l'annotation du membre gauche de l'application d'une règle spécialisée à la production d'un fragment est toujours plus riche que celle de son membre droit. Cette propriété est toujours assurée avec une approximation uniforme du flot d'information puisque chaque sorte de protéines n'apparaît qu'une seule fois sur



(a) Les règles du modèle.



(b) La carte de contacts annotée.



(c) Un dépliage de la carte de contacts annoté.

Figure 4.15: Un modèle avec un flot d'information contextuel. Ce modèle décrit l'évolution des configurations d'une protéine à trois sites g, h et d respectivement représentés à gauche, en haut et à droite de l'occurrence de la protéine. En 4.15(a) sont données les règles de ce modèle. En particulier, lorsque le site g est phosphorylé, le site d peut devenir phosphorylé uniquement si le site h l'est aussi. Par contre, lorsque le site g n'est pas phosphorylé, le site d peut le devenir quel que soit l'état du site g. En 4.15(b) est représenté la carte de contacts annotée de ce modèle. En 4.15(c) est dessiné une annotation d'un dépliage de la carte de contacts qui distinguent les occurrences de la protéine selon l'état de phosphorylation du site g.

la carte de contacts.

Les analyses non-uniforme offrent une hiérarchie d'approximations pour réduire la sémantique différentielle des modèles écrits en Kappa. Il n'existe pour l'instant pas de méthodes pour savoir quels contextes doivent être distingués et ainsi, choisir le meilleur compromis entre précision et complexité de l'analyse.

Exemple 4.6.1 *Le modèle défini par l'ensemble de règles donné en figure 4.15(a) décrit l'évolution des occurrences d'une protéine avec trois sites, nommés g, h et d. L'élément le plus intéressant est la phosphorylation du site d. Celle-ci peut avoir lieu lorsque le site g n'est pas phosphorylé ou lorsque les deux sites g et h sont tous deux phosphorylés.*

Une approximation uniforme ne donne pas d'information assez précise (voir en figure 4.15(b)). Puisqu'il n'y a qu'une seule occurrence de la protéine dans la carte de contacts, le flot d'information est résumé de manière conservative en indiquant un flot d'information potentiel du site h vers le site d, que le site g soit phosphorylé ou non. Une meilleure approximation du flot d'information est obtenue en distinguant les deux états de phosphorylation potentiels du site g (voir en figure 4.15(c)). Ceci permet de noter l'absence de flot d'information du site h vers le site d lorsque le site g n'est pas phosphorylé.

Des approches analytiques peuvent trouver de meilleurs changements de variables pour réduire la sémantique différentielle des réseaux réactionnels. Toutefois, comme l'espace des changements de variables est vaste, elles se restreignent le plus souvent à l'exploration d'un sous-ensemble de celui-ci, comme celui des bisimulations qui sont induites par une relation d'équivalence entre les différentes configurations d'espèces biochimiques du modèle [34, 36]. L'approche présentée ici présente deux avantages. Comme elle s'appuie sur des propriétés structurelles, elle n'a pas besoin de la représentation explicite du réseau réactionnel ou de sa sémantique différentielle initiale. Par ailleurs, les changements de variables qu'elle produit ne sont en général pas exprimables comme une bisimulation induite par une relation d'équivalence entre les différentes configurations d'espèces biochimiques du modèle. Par contre, cette méthode n'offre aucune garantie d'optimalité sur le sous-ensemble des changements de variables considéré. Par ailleurs, les réductions de modèles par les approches analytiques ne sont valables que pour un jeu de constantes de réactions données, alors que la méthode proposée dans cette partie donne une réduction

de modèles qui est valable quelque soit les constantes des règles d'interactions. En contre-partie, cette méthode ne peut pas exploiter les relations numériques éventuelles entre les contantes des règles d'interactions.

Les méthodes de réduction exactes sont par définition limitées par le critère de correction qui est trop exigeant. De plus, elles n'offrent aucune marge de manœuvre sur le choix des observables. Une alternative est de considérer des réductions numériquement approchées. En fixant à la main un ensemble d'observables, il est possible d'abstraire l'état du système par une hyper-boîte qui encadre la valeur de chaque observable entre les bornes d'un intervalle. Le modèle réduit consiste alors en une équation différentielle sur les bornes des différents intervalles ou, d'un point de vue géométrique, sur les coordonnées des hyper-faces de l'hyper-boîte. Ces équations sont obtenues en majorant la dérivée de chaque observable au voisinage de l'hyper-face supérieure correspondante et en minorant la dérivée de chaque observable au voisinage de l'hyper-face inférieure correspondante. Cette approche a été utilisée pour retranscrire dans un cadre formel des méthodes de troncation [123] et des méthodes de tropicalisation [7] tout en fournissant, à chaque instant, un encadrement correct de la valeur des observables, ce qui va bien au delà des résultats de convergence asymptotiques habituels.

Chapitre 5

Flot d'information dans la sémantique stochastique d'un modèle Kappa

Les modèles d'interaction entre protéines souffrent d'une très grande complexité combinatoire. Les protéines peuvent se lier entre elles et modifier leurs états, ce qui conduit potentiellement à la formation de très grandes espèces biochimiques. Ceci empêche toute description extensionnelle des systèmes dynamiques engendré par ces modèles, que ce soit dans un cadre différentiel ou stochastique.

Le chapitre précédent portait sur une approche qui permettait de réduire le nombre de variables nécessaire pour décrire la dynamique de ces modèles dans le cadre différentiel. En suivant le flot d'information potentiel entre les différentes parties des espèces biochimiques, elle consiste à identifier quelles corrélations entre états de sites d'interaction ont une importance sur la dynamique du système, puis à oublier les autres en découpant les espèces biochimiques en petite unité d'information, appelées fragments. Alors qu'elle donne des résultats prometteurs dans le cadre différentiel, le présent chapitre montre au contraire qu'il est beaucoup plus difficile de réduire la dimension des modèles stochastiques engendrés par un modèle de réécriture de graphes à sites.

Ce chapitre commence par la description formelle de la sémantique stochastique d'un modèle Kappa. En Sect. 5.2 sont fournis trois exemples pour illustrer en quoi l'approche décrite dans le chapitre 4 pour la réduction des systèmes différentiels engendrés ne peut pas s'appliquer directement pour réduire les systèmes stochastiques sous-jacents. Enfin, en Sect. 5.3, est décrit la méthode pour réduire la sémantique stochastique des modèles écrits en Kappa. Il s'agit d'une restriction de celle utilisée pour la sémantique différentielle.

Remerciement

Ces travaux ont été effectués en collaboration avec Tatjiana Petrov, pendant sa thèse [116]. Les principales publications les concernant sont les suivantes [76, 72].

5.1 Sémantique stochastique

La sémantique stochastique d'un modèle de réécriture de graphes à sites décrit la distribution de probabilité des comportements possibles de ce modèle au cours du temps. Elle agit sur des états qui sont des graphes à sites entièrement spécifiés, qui peuvent donc être vus comme des multi-ensembles de configurations d'espèces biochimiques.

La sémantique stochastique se décrit à différents niveaux d'abstraction. Dans cette partie, sont détaillées les sémantiques les plus utilisées. La sémantique de traces définit l'exécution du système par une succession d'états reliés par des pas de calculs. Ceux-ci interviennent à des moments précis de l'exécution. Cette sémantique associe des probabilités à des ensembles de traces particuliers [76]. L'équation maîtresse décrit, elle, l'évolution temporelle, de la probabilité, pour chaque état potentiel, d'être dans cet état [114]. Elle est caractérisée par une équation différentielle qui comporte une variable par état potentiel. L'équation maîtresse peut être vue comme une abstraction de la sémantique de traces. Contrairement à cette dernière, elle ne permet pas d'observer des propriétés portant sur différents instants de l'exécution du système. Par exemple, il n'est pas possible à partir de la solution de l'équation maîtresse de calculer la probabilité que le système soit dans l'état q_3 à l'instant t_3,

sachant qu'il était dans l'état q_2 à l'instant t_2 et dans l'état q_1 à l'instant t_1. Or cette probabilité conditionnelle peut être calculée à partir de la sémantique de traces. Par contre, la solution de l'équation maîtresse permet de calculer très facilement des espérances de nombre d'occurrences de configurations d'espèces biochimiques conditionnées ou non par des propriétés sur l'état courant du système.

Comme il a été vu en exemple 2.9.1 page 23, un ensemble de règles de réécritures de graphes à sites engendre un réseau de réactions biochimiques. Il suffit donc de définir la sémantique de traces et l'équation maîtresse des réseaux de réactionnels. Il est toutefois important de noter que la sémantique de traces peut être échantillonnée par simulation directement au niveau des graphes à sites. Ceci évite d'avoir à calculer le réseau réactionnel et donc l'ensemble des configurations d'espèces biochimiques potentielles. Diverses structures de données ont été proposées pour représenter et mettre à jour efficacement l'ensemble des pas de calculs potentiels lors des simulations stochastiques [54, 17].

5.1.1 Sémantique de traces et simulation

La sémantique de traces s'obtient en associant à chaque réaction, une loi exponentielle qui définit pour chaque intervalle de temps, la probabilité que cette réaction s'applique dans cet intervalle de temps si aucune autre réaction n'est intervenue auparavant. La loi exponentielle est paramétrée par une constante, appelée la propensité de la réaction, qui sera alors la constante d'interaction de cette réaction multipliée par son nombre d'applications potentielles dans l'état du système. Le nombre d'applications d'une réaction est égal au produit, pour chaque réactif, de l'expression $\frac{n!}{(n-\alpha)!}$, où n désigne le nombre d'occurrences de ce réactif dans l'état du système et α le coefficient stœchiométrique de ce réactif dans le membre gauche de la réaction. Il existe en fait n possibilités pour choisir la première occurrence. Puis comme il faut choisir des occurrences distinctes, il reste $n-1$ possibilités pour la seconde, et ainsi de suite. Au total, il y aura donc $n \cdot (n-1) \cdots (n-\alpha+2) \cdot (n-\alpha+1)$ possibilités, ce qui est égal au rapport en question.

Il existe en fait plusieurs conventions pour tenir compte des coefficients stœchiométiques dans les membres gauches d'une règle. Il est possible de considérer que le choix porte sur un ensemble ou une liste de α occurrences. Dans le premier cas, il faut diviser la constante de réaction par le nombre de permutations correspondant, c'est à dire $\alpha!$. C'est la deuxième possibilité qui a été adoptée ici. Les constantes de réactions ne sont donc pas corrigées. Le choix de telle ou telle convention n'a aucune incidence sur le cadre de travail. Il suffit en effet d'inclure les coefficients correctifs éventuels dans les constantes de réactions.

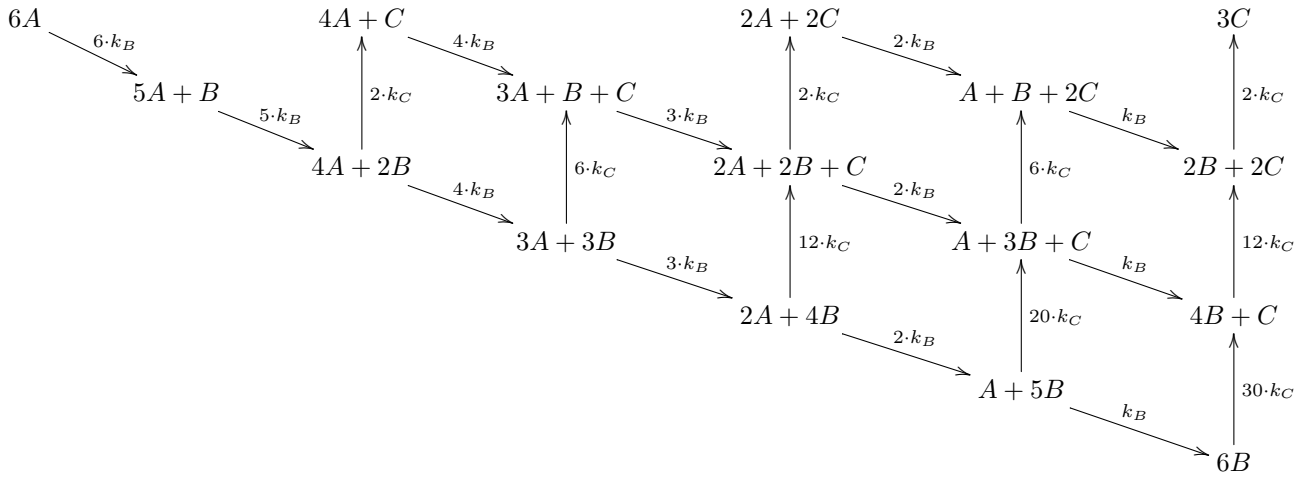
Exemple 5.1.1 *L'exemple introduit en figure 3.1(a) est maintenant utilisé pour illustrer la construction de la sémantique de traces. Il comporte trois espèces biochimiques A, B et C et les deux réactions données en figure 5.1(a). Ainsi, une occurrence de l'espèce biochimique A peut se transformer en une occurrences de l'espèce biochimique B avec une constante de réaction k_B et deux occurrences de l'espèces biochimiques B peuvent se transformer en une occurrence de l'espèce biochimique C avec une constante de réaction k_C. Dans un état donné comportant n_A occurrences de l'espèce biochimique A et n_B occurrences de l'espèce biochimique B, la propensité de la première réaction est donnée par l'expression $k_B \cdot n_A$ et celle de la seconde par l'expression $k_c \cdot n_B \cdot (n_B - 1)$.*

En figure 5.1(b) est donné le système de transitions associé à ces deux réactions, en partant d'un état comportant 6 *occurrences de l'espèce biochimique A. Chaque transition est étiquetée par sa propensité.*

Pour définir l'exécution stochastique d'un modèle, if faut spécifier selon quelles lois de probabilité la prochaine transition est choisie et à quel moment elle intervient. Dans un état donné, la transition suivante est choisie parmis toutes celles possibles avec une probabilité proportionnelle à sa propensité. Ainsi, en appelant l'activité du système, la somme des propensités des transitions qui s'appliquent dans son état actuel, une transition est choisie avec la probabilité égale au quotient entre sa propensité et l'activité du système. Grâce aux propriétés de distributivité des lois exponentielles, le prochain événement interviendra à un instant déterminé suivant la fonction de densité de probabilité exponentielle avec pour paramètre λ l'activité du système. Ainsi la probabilité que la prochaine transition intervienne dans un délai compris entre un temps t_1 et t_2 sera égale à $e^{-\lambda \cdot t_1} \cdot \left(1 - e^{-\lambda \cdot (t_2 - t_1)}\right)$. Ceci permet de définir l'algorithme de la prochaine réaction de Doob-Gillespie [86]. C'est cet algorithme qui est utilisé pour échantillonner les traces de la sémantique stochastique dans le simulateur des modèles Kappa [55]. Par ailleurs, ceci permet également de définir la probabilité de certains ensembles de traces. Il n'est, en effet, pas possible de définir la probabilité d'une trace précise dans laquelle les moments exacts où chaque transition intervient sont entièrement déterminés. Une telle trace aurait une probabilité nulle. Il faut donc considérer des traces dans lesquelles le délai entre deux transitions se situe dans un intervalle non dégénéré (c'est à dire qui contient au moins deux points distincts et donc, tous les points compris entre ces deux

$$\begin{cases} r_B \ : \ A \to B \ \ @\, k_B \\ r_C \ : \ 2B \to C \ \ @\, k_C \end{cases}$$

(a) Un réseau réactionnel.



(b) Système de transitions sous-jacent.

Figure 5.1: Un réseau réactionnel et sa sémantique stochastique décrite sous la forme d'un système de transition pondéré. En 5.1(a) un réseau réactionnel formé de deux réactions. La première, nommée r_B, permet de transformer une occurrence de l'espèce biochimique A en une occurrence de l'espèce biochimique B, la seconde, appelée r_C, permet de transformer deux occurrences de l'espèce biochimique B en une occurrence de l'espèce biochimique C. La restriction de l'ensemble de toutes les transitions possibles aux états qui sont atteignables à partir d'un état initial formé de six occurrences de la protéine A est dessinée en 5.1(b) sous la forme d'un système de transitions. Chaque transition est étiquetée par sa propensité. La propensité de la première réaction est donnée par l'expression $k_B \cdot n_A$. Celle de la seconde par l'expression $k_c \cdot n_B \cdot (n_B - 1)$.

points). Une telle trace est appelé un *ensemble élémentaire de traces* [76]. Un tel ensemble de traces sera noté de la manière suivante :

$$q_0 \xrightarrow{r_1,I_1} q_1 \dots q_{n-1} \xrightarrow{r_n,I_n} q_n$$

où n est la longueur des traces dans l'ensemble élémentaire de traces, pour tout entier i entre 0 et n, q_i est un état et pour tout i compris entre 1 et n, $q_{i-1} \xrightarrow{r_i,I_i} q_i$ est une transition obtenue en appliquant la règle nommée r_i avec un délai depuis l'application de la règle précédente compris dans l'intervalle de temps I_i.

La probabilité d'un tel ensemble de traces est obtenue en faisant le produit entre la probabilité de commencer dans l'état q_0 dans la distribution des états initiaux (qui est un paramètre du modèle) et, pour chaque transition, du produit entre la probabilité d'appliquer la réaction r_i et la probabilité que la prochaine réaction intervienne dans l'intervalle de temps I_i.

Exemple 5.1.2 *L'exemple 5.1.1 se poursuit par le calcul la probabilité de l'ensemble élémentaire de traces suivant :*

$$\langle 6,0,0\rangle \xrightarrow{r_B,[2,3]} \langle 5,1,0\rangle \xrightarrow{r_B,[1,4]} \langle 4,2,0\rangle \xrightarrow{r_C,[2,5]} .\langle 4,0,1\rangle$$

Dans cet ensemble de traces, le système débute dans l'état formé uniquement de 6 *occurrences de l'espèce biochimique A. Ensuite la réaction r_B est appliquée dans un intervalle de temps compris entre* 2 *unités de temps et* 3 *unités de temps. Puis la réaction r_B est de nouveau appliquée. Le délai ente les deux applications de la règle r_B est compris entre* 1 *unité de temps et* 4*. Enfin, la réaction r_C est appliquée. Le délai entre la deuxième application de la règle r_B et celle-ci est compris entre* 2 *unités de temps et* 5*.*

En supposant que la distribution des états initiaux du système associe la probabilité 1 *à l'état formé de* 6 *occurrences de l'espèce biochimique A, la probabilité de l'ensemble élémentaire de traces ci-dessus est donnée par le calcul suivant :*

$$1 \cdot 1 \cdot e^{-6\cdot k_B \cdot 2} \cdot (1 - e^{-6\cdot k_B \cdot 1}) \cdot 1 \cdot e^{-5\cdot k_B \cdot 1} \cdot (1 - e^{-5\cdot k_B \cdot 3}) \cdot \frac{2 \cdot k_C}{4 \cdot k_B + 2 \cdot k_C} \cdot e^{-(4\cdot k_B + 2\cdot k_C)\cdot 2} \cdot (1 - e^{-(4\cdot k_B + 2\cdot k_C)\cdot 3})$$

Dans ce dernier, chaque facteur est représenté explicitement. De gauche à droite, le facteur 1 *correspond au choix de l'état initial. Le second facteur* 1 *correspond au choix de la réaction r_B qui est la seule à pouvoir s'appliquer. Ensuite vient le facteur pour définir quand la première transition intervient. Il dépend de l'activité du système $6 \cdot k_B$, de la borne inférieure* 2 *de l'intervalle de temps et de sa taille* 1*. Le facteur* 1 *suivant correspond au choix de la réaction r_B, qui est là encore la seule à pouvoir s'appliquer. Le facteur temporel est défini à partir de l'activité $5 \cdot k_B$, de la borne inférieure* 1 *de l'intervalle de temps et de sa taille* 3*. Enfin, le facteur $\frac{2\cdot k_C}{4\cdot k_B + 2\cdot k_C}$ correspond au choix de la réaction r_C dont la propensité est $2 \cdot k_C$ pour une activité égale à $4 \cdot k_B + 2 \cdot k_C$. Le facteur temporel fait également intervenir la borne inférieure* 2 *de l'intervalle de temps et sa taille* 3*.*

L'expression se simplifie en la suivante :

$$\frac{2 \cdot k_C \cdot e^{-(25\cdot k_B + 4\cdot k_C)} \cdot (1 - e^{-6\cdot k_B}) \cdot (1 - e^{-15\cdot k_B}) \cdot (1 - e^{-(12\cdot k_B + 6\cdot k_C)})}{4 \cdot k_B + 2 \cdot k_C}.$$

5.1.2 Équation maîtresse

L'*équation maîtresse* [106, 68, 64] permet de calculer l'évolution au cours du temps de la probabilité que le système soit dans un état, pour tout état potentiel du système. C'est un système d'équations différentielles qui comporte autant de variables que d'états potentiels. Pour chaque état potentiel du système, q, la variable $P_t(q)$ représente la probabilité que le système soit dans l'état q à l'instant t. Chaque équation est obtenue en considérant sur un temps infinitésimal, la probabilité d'arriver dans un état et la probabilité de le quitter. Plus précisément, la probabilité que le système arrive dans l'état q par une transition de l'état q^{-1} vers l'état q de propensité λ est égale à $\lambda \cdot P_t(q^{-1})$ alors que la probabilité de le quitter est égale à $act(q) \cdot P_t(q)$, où $act(q)$ désigne l'activité du système dans l'état q.

Ainsi, en représentant chaque transition du système par un triplet (q, λ, q') où q est l'état source, q' l'état cible et λ la propensité de cette transition, et en notant T l'ensemble de toutes les transitions du système, l'équation maîtresse est le système d'équations différentielles qui associe à tout état $q^\star$ l'équation différentielle suivante :

$$\frac{dP_t(q^\star)}{dt} = \left(\sum_{(q,\lambda,q') \in T, q' = q^\star} \lambda \cdot P_t(q) \right) - \left(\sum_{(q,\lambda,q') \in T, q = q^\star} \lambda \cdot P_t(q^\star) \right).$$

En particulier, si une transition ne change pas l'état du système, elle contribue positivement et négativement à l'équation maîtresse et ces deux contributions s'annulent. Par ailleurs, dans le second terme de l'équation, il est possible de retrouver l'activité du système en factorisant l'expression $P_t(q^\star)$.

Exemple 5.1.3 *L'équation maîtresse pour l'exemple 5.1.1 est étudiée ici.*

L'état du système est noté $\langle n_A, n_B, n_C\rangle$, où n_A désigne le nombre d'occurrences de l'espèce biochimique A, n_B celui de l'espèce biochimique B et n_C celui de l'espèce biochimique C. Étant donné un état $\langle n_A, n_B, n_C\rangle$, la dérivée $\frac{dP_t(\langle n_A,n_B,n_C\rangle)}{dt}$ de la probabilité que le système soit dans l'état $\langle n_A, n_B, n_C\rangle$ à l'instant t est donnée par l'équation suivante :

$$\frac{dP_t(\langle n_A, n_B, n_C\rangle)}{dt} =$$
$$\begin{cases} k_B{\cdot}(n_A+1){\cdot}P_t(\langle n_A{+}1, n_B{-}1, n_C\rangle) + k_C{\cdot}(n_B+2)\cdot(n_B+1){\cdot}P_t(\langle n_A, n_B{+}2, n_C{-}1\rangle) & \\ \qquad -(k_B\cdot n_A + k_C\cdot n_B\cdot(n_B-1)){\cdot}P_t(\langle n_A, n_B, n_C\rangle) & \text{si } n_B \geq 1 \text{ et } n_C \geq 1; \\ k_C{\cdot}(n_B+2)\cdot(n_B+1){\cdot}P_t(\langle n_A, n_B{+}2, n_C{-}1\rangle) & \\ \qquad -(k_B\cdot n_A + k_C\cdot n_B\cdot(n_B-1)){\cdot}P_t(\langle n_A, n_B, n_C\rangle) & \text{si } n_B = 0 \text{ et } n_C \geq 1; \\ k_B{\cdot}(n_A+1){\cdot}P_t(\langle n_A{+}1, n_B{-}1, n_C\rangle) + k_C{\cdot}(n_B+2)\cdot(n_B+1){\cdot}P_t(\langle n_A, n_B{+}2, n_C{-}1\rangle) & \\ \qquad -(k_B\cdot n_A + k_C\cdot n_B\cdot(n_B-1)){\cdot}P_t(\langle n_A, n_B, n_C\rangle) & \text{si } n_C = 0 \text{ et } n_B \geq 1; \\ -(k_B\cdot n_A + k_C\cdot n_B\cdot(n_B-1)){\cdot}P_t(\langle n_A, n_B, n_C\rangle) & \text{si } n_B = 0 \text{ et } n_C = 0. \end{cases}$$

Dans cette équation, les flux positifs liés à la première réaction et à la seconde, sont conditionnées par le fait que les produits correspondants sont présent dans l'état du système. En fait, dans le cas contraire, il est impossible que l'état actuel soit l'état d'une transition appliquant la réaction correspondante. Les flux négatifs, eux, ne sont pas soumis à des conditions car ils comportent des facteurs qui s'annulent quand l'état présent ne contient pas assez de réactifs. Le système d'équations différentielles obtenu en instanciant cette équation à tous les états du système est donné en figure 5.2.

Lorsque le système contient suffisamment peu d'états potentiels, l'équation maîtresse peut être intégrée numériquement. Il est alors possible de calculer l'évolution temporelle de l'espérance du nombre d'occurrences d'une espèce biochimique ou même l'évolution de l'espérance conditionnelle du nombre d'occurrences d'une espèce biochimique sachant que l'état vérifie une certaine condition.

Exemple 5.1.4 *Pour conclure l'exemple 5.1.1, l'espérance $E_t(n_B)$ du nombre d'occurrences de l'espèce biochimique B à l'instant t est donnée par l'expression suivante :*

$$E_t(n_B) = \sum_{\langle n_A,n_B,n_C\rangle\in\mathcal{Q}} n_B{\cdot}P_t(\langle n_A, n_B, n_C\rangle)$$

dans laquelle $\mathcal{Q}$ représente l'ensemble de tous les états potentiels du système.

Par ailleurs, l'espérance conditionnelle $E_t(n_B \mid n_C = 0)$ à l'instant t du nombre d'occurrences de l'espèce biochimique B sachant qu'il n'y a pas d'occurrences de l'espèce biochimique n_C dans l'état du système, est, lorsqu'elle est bien définie, donnée par l'expression suivante :

$$E_t(n_B \mid n_C = 0) = \frac{\sum_{\langle n_A,n_B,n_C\rangle\in\mathcal{Q},n_C=0} n_B{\cdot}P_t(\langle n_A, n_B, n_C\rangle)}{\sum_{\langle n_A,n_B,n_C\rangle\in\mathcal{Q},n_C=0} P_t(\langle n_A, n_B, n_C\rangle)}$$

Dans l'expression précédente, le dénominateur correspond à la probabilité d'être dans un état qui ne contient pas d'occurrences de l'espèce biochimique C. L'espérance conditionnelle n'est définie que lorsque cette probabilité est non nulle.

5.1.3 Retour sur les pas de calculs

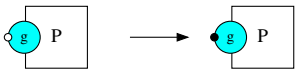
La simulation d'un modèle Kappa opère directement par réécriture du graphe qui représente l'état du système, sans avoir à considérer le réseau de réactions sous-jacent [54, 17]. S'il est possible de passer par l'ensemble de réactions qui est induit par un modèle de réécriture de graphes à sites, il est cependant parfois plus pratique de définir les transitions, ou de *pas de calcul*, directement en Kappa.

$$
\begin{aligned}
\frac{dP_t(\langle 0,0,3\rangle)}{dt} &= 2{\cdot}k_C{\cdot}P_t(\langle(0,2,2)\rangle)\\
\frac{dP_t(\langle 0,2,2\rangle)}{dt} &= k_B{\cdot}P_t(\langle(1,1,2) + 12{\cdot}k_C{\cdot}P_t(\langle 0,4,1\rangle) - 2{\cdot}k_C{\cdot}P_t(\langle(0,2,2)\rangle)\\
\frac{dP_t(\langle 0,4,1\rangle)}{dt} &= k_B{\cdot}P_t(\langle(1,3,1)\ 30{\cdot}k_C{\cdot}P_t(\langle 0,6,0\rangle) - 12{\cdot}k_C{\cdot}P_t(\langle 0,4,1\rangle)\\
\frac{dP_t(\langle 0,6,0\rangle)}{dt} &= k_B{\cdot}P_t(\langle(1,5,0)\rangle) - 30{\cdot}k_C{\cdot}P_t(\langle 0,6,0\rangle)\\
\frac{dP_t(\langle 1,1,2\rangle)}{dt} &= 2{\cdot}k_B{\cdot}P_t(\langle(2,0,2)\rangle) + 6{\cdot}k_C{\cdot}P_t(\langle(1,3,1)\ - k_B{\cdot}P_t(\langle(1,1,2)\rangle)\\
\frac{dP_t(\langle 1,3,1\rangle)}{dt} &= 2{\cdot}k_B{\cdot}P_t(\langle(2,2,1)\rangle)20{\cdot}k_C{\cdot}P_t(\langle(1,5,0)\rangle) - (k_B + 6{\cdot}k_C){\cdot}P_t(\langle(1,3,1)\rangle)\\
\frac{dP_t(\langle 1,5,0\rangle)}{dt} &= 2{\cdot}k_B{\cdot}P_t(\langle(2,4,0)\rangle) - (k_B + 20{\cdot}k_C){\cdot}P_t(\langle(1,5,0)\rangle)\\
\frac{dP_t(\langle 2,0,2\rangle)}{dt} &= 2{\cdot}k_C{\cdot}P_t(\langle(2,2,1)\rangle) - 2{\cdot}k_B{\cdot}P_t(\langle(2,0,2)\rangle)\\
\frac{dP_t(\langle 2,2,1\rangle)}{dt} &= 3{\cdot}k_B{\cdot}P_t(\langle(3,1,1) + 12{\cdot}k_C{\cdot}P_t(\langle(2,4,0)\rangle) - (2{\cdot}k_B + 2{\cdot}k_C){\cdot}P_t(\langle(2,2,1)\rangle)\\
\frac{dP_t(\langle 2,4,0\rangle)}{dt} &= 3{\cdot}k_B{\cdot}P_t(\langle(3,3,0)\rangle) - (2{\cdot}k_B + 12{\cdot}k_C){\cdot}P_t(\langle(2,4,0)\rangle)\\
\frac{dP_t(\langle 3,1,1\rangle)}{dt} &= 4{\cdot}k_B{\cdot}P_t(\langle(4,0,1) + 6{\cdot}k_C{\cdot}P_t(\langle(3,3,0)\rangle) - 3{\cdot}k_B{\cdot}P_t(\langle(3,1,1)\rangle)\\
\frac{dP_t(\langle 3,3,0\rangle)}{dt} &= 4{\cdot}k_B{\cdot}P_t(\langle(4,2,0)\rangle) + -(3{\cdot}k_B + 6{\cdot}k_C){\cdot}P_t(\langle(3,3,0)\rangle)\\
\frac{dP_t(\langle 4,0,1\rangle)}{dt} &= 2{\cdot}k_C{\cdot}P_t(\langle(4,2,0)\rangle) - 4{\cdot}k_B{\cdot}P_t(\langle(4,0,1)\rangle)\\
\frac{dP_t(\langle 4,2,0\rangle)}{dt} &= 5{\cdot}k_B{\cdot}P_t(\langle(5,1,0) - (4{\cdot}k_B + 2{\cdot}k_C){\cdot}P_t(\langle(4,2,0)\rangle)\\
\frac{dP_t(\langle 5,1,0\rangle)}{dt} &= 6{\cdot}k_B{\cdot}P_t(\langle(6,0,0)\rangle) - 5{\cdot}k_B{\cdot}P_t(\langle(5,1,0)\rangle)\\
\frac{dP_t(\langle 6,0,0\rangle)}{dt} &= -6{\cdot}k_B{\cdot}P_t(\langle(6,0,0)\rangle)
\end{aligned}
$$

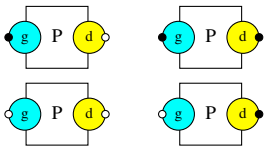
Figure 5.2: Équation maîtresse pour le réseau réactionnel de la figure 5.1(a). Chaque transition du système de transitions de la figure 5.1(b) apporte une contribution positive pour la cible de la transition, qui correspond à la probabilité d'entrer dans un nouvel état sur un temps infinitésimal et une contribution négative pour la source de la transition, qui correspond à la probabilité de quitter l'état courant sur un temps infinitésimal. Chaque contribution dépend de la probabilité d'être dans l'état source de la transition correspondante.

Ainsi, étant donnée une règle de réécriture, un pas de calcul impliquant cette règle peut se définir en appliquant une réaction obtenue comme raffinement de cette règle. Mais un tel pas de calcul peut se définir directement en raffinant la règle jusqu'à ce que le membre gauche de la règle raffinée coïncide avec l'état source du pas de calcul. Le membre droit de la règle raffinée constitue alors l'état cible de la transition.

Exemple 5.1.5 *Pour reprendre l'exemple donné en figure 2.13, il est possible d'appliquer la règle suivante :*

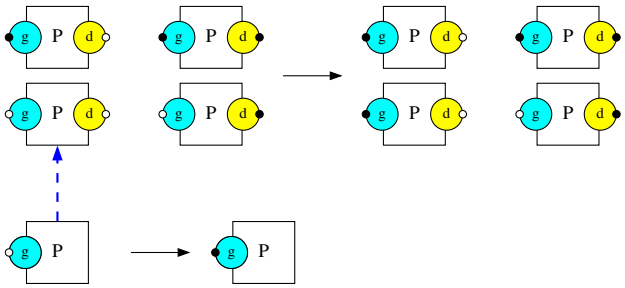


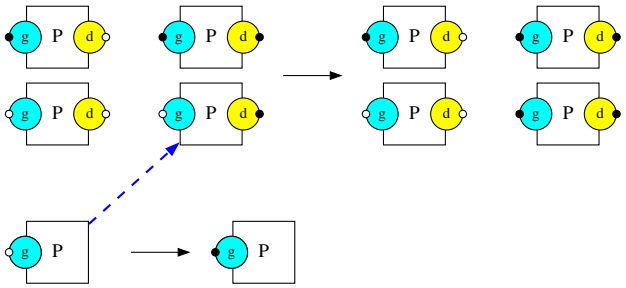
pour phosphoryler le site g d'une occurrence de configurations de la protéine P dans l'état suivant :



formé de quatre occurrences de la protéine P, chacune dans l'une des quatre configurations potentielles de la protéine.

Pour ce faire, il suffit de choisir un plongement entre le membre gauche de la règle et le graphe à sites qui constitue l'état du système. Il y a deux possibilités qui induisent les deux pas suivants :





Ces pas de calculs auraient pu être définis à partir du réseau réactionnel sous-jacent en appliquant les deux règles-réactions suivantes, selon que la règle est appliquée à une occurrence de configurations de la protéine P dans laquelle le site d est phosphorylé, ou non.

5.2 Cas d'étude

Trois cas d'études sont considérés pour étudier s'il est possible d'utiliser le flot d'information entre les sites d'interaction des configurations des espèces biochimiques pour simplifier la sémantique stochastique de modèles de réécriture de graphes à sites. Le premier, qui est décrit en Sect. 5.2.1, fait intervenir une protéine dont les occurrences peuvent se lier indépendamment aux occurrences de deux autres. La première protéine peut alors

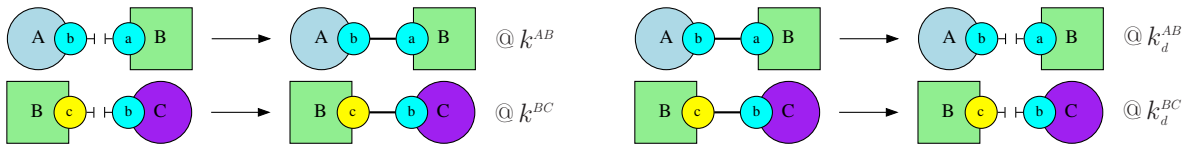


Figure 5.3: Les règles d'interaction d'association et de dissociation dans le modèle des deux liaisons indépendantes.

être vue comme deux morceaux entièrement indépendant, ce qui conduit à une réduction du modèle correcte dans le cadre stochastique. Dans le second, donné en Sect. 5.2.2, il est montré que des corrélations sont souvent nécessaires pour simuler fidèlement les systèmes stochastiques. Dans cette exemple, une règle de dissociation ne peut être simplifiée sans connaître précisément la distribution des différents configurations des occurrences de protéines liées par la liaison que cette règle supprime. Dans le cadre différentiel, cette information pouvait être oubliée tout en obtenant une réduction exacte du système engendré. Enfin, le troisième exemple, qui est expliqué en Sect. 5.2.3, montre l'existence de flot d'information entre différentes composantes connexes du membre gauche d'une règle, ce qui une fois encore interdit certaines réductions qui sont pourtant correctes dans le cadre différentiel.

5.2.1 Un exemple d'indépendance entre deux liaisons

Voici un modèle dans lequel chaque occurrence d'une protéine peut être découpée en deux morceaux indépendants. La dimension de l'espace d'états du système stochastique sous-jacent peut alors être réduite en conséquence.

Soit une protéine B munie de deux sites d'association a et c. Dans chaque occurrence de la protéine B, le site a peut se lier au site b des occurrences d'une autre protéine, appelée A, alors que le c peut se lier au site b des occurrences d'une troisième protéine, appelée C. Les règles d'association et de dissociation sont dessinées en Fig. 5.3. Les constantes d'interaction pour les règles d'association et de dissociation ne dépendent pas du fait que l'occurrence de la protéine B soit liée, ou non, sur son autre site d'association. Ce point est essentiel pour justifier la correction de la réduction du modèle qui va suivre.

5.2.1.1 Réduction de la sémantique différentielle

Avant de regarder comment réduire la sémantique stochastique engendrée par ces règles, nous vérifions qu'il est facile de réduire la sémantique différentielle. Cette dernière est définie par le système d'équations différentielles ordinaires suivant :

$$\begin{aligned}
\frac{d[A]}{dt} &= k_d^{AB}\left([AB]+[ABC]\right) - k^{AB}\left([B]+[BC]\right)[A] \\
\frac{d[C]}{dt} &= k_d^{BC}\left([BC]+[ABC]\right) - k^{BC}\left([B]+[AB]\right)[C] \\
\frac{d[B]}{dt} &= k_d^{AB}[AB] + k_d^{BC}[BC] - \left(k^{AB}[A] + k^{BC}[C]\right)[B] \\
\frac{d[AB]}{dt} &= k^{AB}[A][B] + k_d^{BC}[ABC] - \left(k_d^{AB} + k^{BC}[C]\right)[AB] \\
\frac{d[BC]}{dt} &= k_d^{AB}[ABC] + k^{BC}[B][C] - \left(k_d^{BC} + k^{AB}[A]\right)[BC] \\
\frac{d[ABC]}{dt} &= k^{AB}[A][BC] + k^{BC}[AB][C] - \left(k_d^{AB} + k_d^{BC}\right)[ABC].
\end{aligned}$$

Chaque espèce biochimique est représentée par la liste des occurrences des protéines qui la constituent. Ainsi, les notations A, B et C représentent les protéines correspondantes, alors que les notations AB, BC, ABC représentent les espèces biochimiques formées d'une occurrence de la protéine B liée à une occurrence de la protéine A, d'une occurrence de la protéine B liée à une occurrence de la protéine C et d'une occurrence de la protéine B liée à une occurrence de la protéine A et à une occurrence de la protéine C.

Ignorer la corrélation entre l'état des deux sites d'associations des occurrences de la protéine B, revient à considerer les équation suivantes :

$$\begin{aligned}
\frac{d[A]}{dt} &= k_d^{AB}[AB?] - k^{AB}[A][B?] \\
\frac{d[B?]}{dt} &= k_d^{AB}[AB?] - k^{AB}[A][B?] \\
\frac{d[AB?]}{dt} &= k^{AB}[A][B?] - k_d^{AB}[AB?] \\
\frac{d[C]}{dt} &= k_d^{BC}[?BC] - k^{BC}[?B][C] \\
\frac{d[?B]}{dt} &= k_d^{BC}[?BC] - k^{BC}[?B][C] \\
\frac{d[?BC]}{dt} &= k^{BC}[?B][C] - k_d^{BC}[?BC]
\end{aligned}$$

où[1] $[AB?] := [AB] + [ABC]$, $[B?] := [B] + [BC]$, $[?BC] := [BC] + [ABC]$, et $[?B] := [B] + [AB]$, qui sont également satisfaites.

5.2.1.2 Réduction de l'équation maîtresse

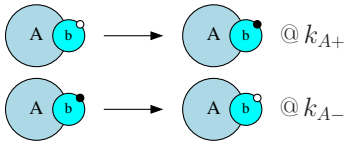
Il est tentant d'essayer d'oublier la corrélation éventuelle entre l'état d'association des deux sites des occurrences de la protéine B également dans le système stochastique engendré par les règles de cet exemple. Pour cela, il est possible de regarder ce qui se passe au niveau de l'évolution de la distribution des différents états que le système peut prendre. L'état du système peut être représenté par un sextuplet d'entiers naturels $\langle n_A, n_B, n_C, n_{AB}, n_{BC}, n_{ABC}\rangle$ où la composante n_X représente le nombre d'occurrences de l'espèce biochimique X dans cet état, pour n'importe quelle espèce biochimique X du modèle. La probabilité $P_t(\sigma)$ que le système soit dans l'état σ au temps t est alors donnée par l'équation maîtresse suivante :

$$\begin{aligned}
\frac{dP_t(\langle n_A, n_B, n_C, n_{AB}, n_{BC}, n_{ABC}\rangle)}{dt} = \quad & k^{AB}(n_A+1)(n_B+1)P_t(\langle n_A{+}1, n_B{+}1, n_C, n_{AB}{-}1, n_{BC}, n_{ABC}\rangle) \\
& + k_d^{AB}(n_{AB}+1)P_t(\langle n_A{-}1, n_B{-}1, n_C, n_{AB}{+}1, n_{BC}, n_{ABC}\rangle) \\
& + k^{AB}(n_A+1)(n_{BC}+1)P_t(\langle n_A{+}1, n_B, n_C, n_{AB}, n_{BC}{+}1, n_{ABC}{-}1\rangle) \\
& + k_d^{AB}(n_{ABC}+1)P_t(\langle n_A{-}1, n_B, n_C, n_{AB}, n_{BC}{-}1, n_{ABC}{+}1\rangle) \\
& + k^{BC}(n_B+1)(n_C+1)P_t(\langle n_A, n_B{+}1, n_C{+}1, n_{AB}, n_{BC}{-}1, n_{ABC}\rangle) \\
& + k_d^{BC}(n_{BC}+1)P_t(\langle n_A, n_B{-}1, n_C{-}1, n_{AB}, n_{BC}{+}1, n_{ABC}\rangle) \\
& + k^{BC}(n_{AB}+1)(n_C+1)P_t(\langle n_A, n_B, n_C{+}1, n_{AB}{+}1, n_{BC}, n_{ABC}{-}1\rangle) \\
& + k_d^{BC}(n_{ABC}+1)P_t(\langle n_A, n_B, n_C{-}1, n_{AB}{-}1, n_{BC}, n_{ABC}{+}1\rangle) \\
& - k^{AB}n_A(n_B+n_{BC})P_t(\langle n_A, n_B, n_C, n_{AB}, n_{BC}, n_{ABC}\rangle) \\
& - k_d^{AB}(n_{AB}+n_{ABC})P_t(\langle n_A, n_B, n_C, n_{AB}, n_{BC}, n_{ABC}\rangle) \\
& - k^{BC}(n_B+n_{AB})n_CP_t(\langle n_A, n_B, n_C, n_{AB}, n_{BC}, n_{ABC}\rangle) \\
& - k_d^{BC}(n_{BC}+n_{ABC})P_t(\langle n_A, n_B, n_C, n_{AB}, n_{BC}, n_{ABC}\rangle)
\end{aligned}$$

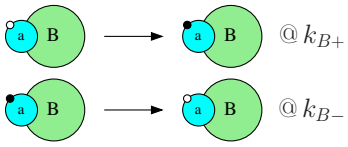
Afin d'oublier la corrélation entre l'état des deux sites d'associations des occurrences de la protéine B, chaque état $\sigma = \langle n_A, n_B, n_C, n_{AB}, n_{BC}, n_{ABC}\rangle$ est abstrait en deux triplets $\beta^A(\sigma) := \langle n_A, n_B + n_{BC}, n_{AB} + n_{ABC}\rangle$ et $\beta^C(\sigma) := \langle n_C, n_B + n_{AB}, n_{BC} + n_{ABC}\rangle$. Ces deux triplets projettent l'état du système σ en passant sous silence l'état d'association respectivement des sites c et a des occurrences de la protéine B. Les évolutions temporelles des distributions de ces deux projections sont alors notées $P_t^A(\sigma^A)$ et $P_t^C(\sigma^C)$. Ainsi, la quantité $P_t^A(\sigma^A)$ représente la probabilité que le système soit dans un état σ tel que $\beta^A(\sigma) = \sigma^A$ à l'instant t, alors la quantité $P_t^C(\sigma^C)$ représente la probabilité que le système soit dans un état σ tel que $\beta^C(\sigma) = \sigma^C$ à l'instant t. Il est alors possible de vérifier analytiquement que l'évolution de deux distributions de probabilités $P_t^A(\sigma^A)$ et $P_t^C(\sigma^C)$ vérifient les deux équations maîtresses suivantes :

$$\begin{aligned}
\frac{dP_t^A(\langle n_A, n_{B?}, n_{AB?}\rangle)}{dt} = \quad & k^{AB}(n_A+1)(n_{B?}+1)P_t^A(\langle n_A{+}1, n_{B?}{+}1, n_{AB?}{-}1\rangle) \\
& + k_d^{AB}(n_{AB?}+1)P_t^A(\langle n_A{-}1, n_{B?}{-}1, n_{AB?}{+}1\rangle) \\
& - (k^{AB}n_An_{B?} + k_d^{AB}n_{AB?})P_t^A(\langle n_A, n_{B?}, n_{AB?}\rangle)
\end{aligned}$$

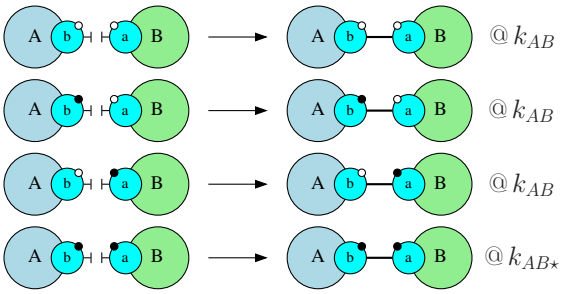
[1] Un point d'interrogation '?' à la gauche (resp. à la droite) de la lettre B indique que de savoir si l'occurrence correspondante de la protéine B est liée, ou non, à une occurrence de la protéine A (resp. C) n'a pas d'importance.



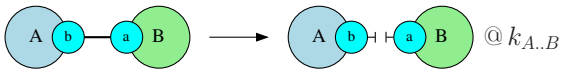
(a) Phosphorylation/déphosphorylation des occurrences de la protéine A.



(b) Phosphorylation/déphosphorylation des occurrences de la protéine B.



(c) Liaison entre les occurrences des protéines A et B.



(d) Déliaison entre les occurrences des protéines A et B.

Figure 5.4: Les règles d'interaction pour le modèle avec une dissociation inconditionnelle. En 5.4(a) et 5.4(b), les occurrences des protéines A et B peuvent se phosphoryler et se déphosphoryler sans conditions. En 5.4(c), elles peuvent se lier. Le taux d'interaction pour lier deux occurrences de configurations de protéines phosphorylées est différent des trois autres cas. Enfin, en 5.4(d) les occurrences de protéines liées peuvent se délier sans conditions.

$$\begin{aligned}\frac{dP_t^C(\langle n_C, n_{?B}, n_{?BC}\rangle)}{dt} = \quad & k^{BC}(n_{?B}+1)(n_C+1)P_t^C(\langle n_C{+}1, n_{?B}{+}1, n_{?BC}{-}1\rangle) \\ & + k_d^{BC}(n_{?BC}+1)P_t^C(\langle n_C{-}1, n_{?B}{-}1, n_{?BC}{+}1\rangle) \\ & - (k^{BC}n_{?B}n_C + k_d^{AB}n_{?BC})P_t^C(\langle n_C, n_{?B}, n_{?BC}\rangle).\end{aligned}$$

De ce fait, il est possible de suivre le comportement du site a de la protéine B tout en oubliant celui du site c, et réciproquement également dans le cadre stochastique

5.2.2 Un exemple avec une dissociation inconditionnelle

Cependant, il est en pratique assez rare de pouvoir réduire la sémantique stochastique d'un modèle en découpant les occurrences des configurations des espèces biochimiques. Ce nouvel exemple contient une règle de dissociation inconditionnelle entre deux occurrences de protéines pouvant prendre plusieurs configurations. Dans la sémantique différentielle, cette règle n'induit pas de flot d'information et le système d'équations différentielles sous-jacent peut être réduit. En stochastique, la corrélation entre l'état des occurrences de protéines qui sont liées entre-elles a un impact sur la distribution de l'état des monomères. Ceci induit un flux d'information qui empêche de réduire de manière exacte le système stochastique sous-jacent.

Soit A et B deux protéines. Chaque occurrence de ces protéines peut être phosphorylée, ou non. De plus, une occurrence de la protéine A peut se lier à une occurrence de la protéine B pour former un dimère. Le comportement de ce modèle peut être décrit par les règles de réécriture données en Fig. 5.4. En particulier, il y a plusieurs constantes d'interaction pour les règles d'association (voir en Fig. 5.4(c)), selon l'état de phosphorylation des occurrences des configurations de protéines qui se lient. Hormis cette règle d'association, les interactions potentielles sont purement locales. Ainsi, les constantes d'interaction pour la phosphorylation des occurrences de la protéine A (voir en Fig. 5.4(a)) et de la protéine B (voir en Fig. 5.4(b)) sont indépendantes du fait que ces occurrences appartiennent à des dimères ou non, et la constante d'interaction pour la dissociation de deux occurrences de protéines est indépendante de l'état de phosphorylation de ces deux occurrences de protéines (voir en Fig. 5.4(c)). Par contre, il existe plusieurs règles d'association (voir en Fig. 5.4(c)), avec des constantes d'interaction pouvant varier selon l'état de phosphorylation des deux occurrences de protéines qui se lient. Par exemple, si la constante d'interaction $k_{AB\star}$ est choisie plus grande que la constante d'interaction k_{AB}, alors la formation de dimère est facilitée quand les deux occurrences des protéines qui se lient sont préalablement phosphorylées. En conséquence, l'état de phosphorylation des occurrences de protéines au sein des dimères n'est pas indépendante. Lorsque l'occurrence de la protéine A est phosphorylée, il y a plus de chance que l'occurrence de la protéine B le soit également.

Il est tentant de vouloir abstraire cette corrélation. Cela reviendrait à oublier quelles occurrences de protéines

$$
\begin{aligned}
\frac{d\,[\circ A\dashv\,]}{dt} &= k_{A-}[\bullet A\dashv\,] + k_{A..B}([\circ AB\circ] + [\circ AB\bullet]) - (k_{A+} + k_{AB}([\vdash B\circ] + [\vdash B\bullet]))\,[\circ A\dashv\,] \\
\frac{d\,[\bullet A\dashv\,]}{dt} &= k_{A+}[\circ A\dashv\,] + k_{A..B}([\bullet AB\circ] + [\bullet AB\bullet]) - (k_{A-} + k_{AB}\,[\vdash B\circ] + k_{AB\star}\,[\vdash B\bullet])\,[\bullet A\dashv\,] \\
\frac{d\,[\vdash B\circ]}{dt} &= k_{B-}[\vdash B\bullet] + k_{A..B}([\circ AB\circ] + [\bullet AB\circ]) - (k_{B+} + k_{AB}([\circ A\dashv\,] + [\bullet A\dashv\,]))\,[\vdash B\circ] \\
\frac{d\,[\vdash B\bullet]}{dt} &= k_{B+}[\vdash B\circ] + k_{A..B}([\circ AB\bullet] + [\bullet AB\bullet]) - (k_{B-} + k_{AB}\,[\circ A\dashv\,] + k_{AB\star}\,[\bullet A\dashv\,])\,[\vdash B\bullet] \\
\frac{d\,[\circ AB\circ]}{dt} &= k_{A-}[\bullet AB\circ] + k_{B-}[\circ AB\bullet] + k_{AB}\,[\circ A\dashv\,]\,[\vdash B\circ] - (k_{A+} + k_{B+} + k_{A..B})\,[\circ AB\circ] \\
\frac{d\,[\bullet AB\circ]}{dt} &= k_{A+}[\circ AB\circ] + k_{B-}[\bullet AB\bullet] + k_{AB}\,[\bullet A\dashv\,]\,[\vdash B\circ] - (k_{A-} + k_{B+} + k_{A..B})\,[\bullet AB\circ] \\
\frac{d\,[\circ AB\bullet]}{dt} &= k_{A-}[\bullet AB\bullet] + k_{B+}[AB] + k_{AB}\,[A]\,[B^\star] - (k_{A+} + k_{B-} + k_{A..B})\,[\circ AB\bullet] \\
\frac{d\,[\bullet AB\bullet]}{dt} &= k_{A+}[\circ AB\bullet] + k_{B+}[\bullet AB\circ] + k_{AB\star}\,[\bullet A\dashv\,]\,[\vdash B\bullet] - (k_{A-} + k_{B-} + k_{A..B})[\bullet AB\bullet].
\end{aligned}
$$

Figure 5.5: Sémantique différentielle pour le modèle de la dissociation inconditionnelle.

$$
\begin{aligned}
\frac{d\,[\circ A\dashv\,]}{dt} &= k_{A-}[\bullet A\dashv\,] + k_{A..B}\,[\circ A-] - (k_{A+} + k_{AB}([\vdash B\circ] + [\vdash B\bullet]))\,[\circ A\dashv\,] \\
\frac{d\,[\bullet A\dashv\,]}{dt} &= k_{A+}[\circ A\dashv\,] + k_{A..B}\,[\bullet A-] - (k_{A-} + k_{AB}\,[\vdash B\circ] + k_{AB\star}\,[\vdash B\bullet])\,[\bullet A\dashv\,] \\
\frac{d\,[\vdash B\circ]}{dt} &= k_{B-}[\vdash B\bullet] + k_{A..B}\,[-B\circ] - (k_{B+} + k_{AB}([\circ A\dashv\,] + [\bullet A\dashv\,]))\,[\vdash B\circ] \\
\frac{d\,[\vdash B\bullet]}{dt} &= k_{B+}[\vdash B\circ] + k_{A..B}\,[-B\bullet] - (k_{B-} + k_{AB}\,[\circ A\dashv\,] + k_{AB\star}\,[\bullet A\dashv\,])\,[\vdash B\bullet] \\
\frac{d\,[\circ A-]}{dt} &= k_{A-}[\bullet A-] + k_{AB}\,[\circ A\dashv\,]\,([\vdash B\circ] + [\vdash B\bullet]) - (k_{A+} + k_{A..B})\,[\circ A-] \\
\frac{d\,[\bullet A-]}{dt} &= k_{A+}[\circ A-] + k_{AB}\,[\bullet A\dashv\,]\,[\vdash B\circ] + k_{AB\star}\,[\bullet A\dashv\,]\,[\vdash B\bullet] - (k_{A-} + k_{A..B})\,[\bullet A-] \\
\frac{d\,[-B\circ]}{dt} &= k_{B-}[-B\bullet] + k_{AB}\,[\vdash B\circ]\,([\circ A\dashv\,] + [\bullet A\dashv\,]) - (k_{B+} + k_{A..B})\,[-B\circ] \\
\frac{d\,[-B\bullet]}{dt} &= k_{B+}[-B\circ] + k_{AB}\,[\circ A\dashv\,]\,[\vdash B\bullet] + k_{AB\star}\,[\bullet A\dashv\,]\,[\vdash B\bullet] - (k_{B-} + k_{A..B})\,[-B\bullet]\,,
\end{aligned}
$$

Figure 5.6: Sémantique différentielle pour le modèle simplifié de la dissociation inconditionnelle.

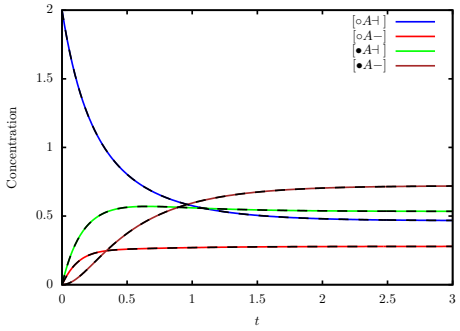


Figure 5.7: Évolution de la concentration des différentes configurations de la protéine A, qu'elle soit phosphorylée, ou non, et quelle soit liée, ou non. Les courbes colorées sont obtenues en simulant numériquement le système d'équations initial, alors que les courbes le sont à partir du système d'équations réduit. Toutes les constantes d'interaction sont fixées à 1, sauf la constante $k_{AB\star}$ qui est fixée à 10. À l'origine, la concentration du monomère de la protéine A dans sa configuration non phosphorylée et celle de celui de la protéine B dans sa configuration non phosphorylée sont toutes deux fixées à 2 ; la concentration des autres configurations de protéines est fixée à 0.

sont liées entre-elles et à suivre indépendamment les occurrences de protéines en ne gardant que leur état de phosphorylation et d'association. Tous les interactions se traduisent directement à ce niveau d'abstraction, sauf celles de dissociation. En effet, une dissociation fait intervenir deux occurrences de protéines liées entre-elles. Comme les liaisons ne sont plus représentées. Du coup, il semble naturel de représenter les règles de dissociation par deux règles indépendantes, l'une pour libérer les occurrences liées de la protéine A et l'autre pour libérer les occurrences liées de le protéine B (ici l'état d'association peut être assimilé à un état interne puisque l'information de savoir quelle occurrence de protéine est liée à quelle autre, n'est pas maintenue).

Quelles sont les conséquences de cette modification des règles du modèles quant à la dynamique des systèmes engendrés ?

5.2.2.1 Réduction de la sémantique différentielle

La sémantique différentielle du modèle initial est définie par le système d'équations donné en figure 5.5. Dans ce système, la concentration de la configuration non phoshporylée du monomère de la protéine A est notée $[\circ A\dashv]$ et la concentration de celle phosphorylée, $[\bullet A\dashv]$. La concentration de la configuration non phosphorylée du monomère de la protéine B est notée $[\vdash B\circ]$ et la concentration de celle phosphorylée, $[\vdash B\bullet]$. La concentration de la configuration du dimère entièrement non phosphorylée est notée $[\circ AB\circ]$, la concentration de la configuration du dimère avec l'occurrence de la protéine A phosphorylée, et non celle de la protéine B, est notée $[\bullet AB\circ]$, la concentration de la configuration du dimère avec l'occurrence de la protéine B phosphorylée, et non celle de la protéine A, est notée $[\circ AB\bullet]$ et la concentration de la configuration du dimère doublement phosphorylée est notée $[\bullet AB\bullet]$.

Le modèle simplifié engendre lui le système différentiel donné en figure 5.6 dans lequel $[\circ A-]$ représente la concentration de la configuration de la protéine A liée et non phosphorylée, $[\bullet A-]$ la concentration de la configuration liée et phosphorylée, $[-B\circ]$ la concentration de la configuration de la protéine B liée et non phosphorylée et $[-B\bullet]$ la concentration de la configuration liée et phosphorylée.

Le système d'équations engendré par le modèle initial et celui-engendré par le modèle simplifié sont liées formellement par les contraintes suivantes :

$$\begin{cases} [\circ A-] := [AB] + [\circ AB\bullet]\,, \\ [\bullet A-] := [\bullet AB\circ] + [\bullet AB\bullet]\,, \\ [-B\circ] := [AB] + [\bullet AB\circ]\,, \\ [-B\bullet] := [\circ AB\bullet] + [\bullet AB\bullet]\,. \end{cases}$$

qui sont obtenues en prenant la définition extensionnelle des parties des configurations des espèces biochimiques.

$$
\begin{aligned}
&\frac{dP_t(\langle n_A, n_{A^\star}, n_B, n_{B^\star}, n_{\circ AB\circ}, n_{\bullet AB\circ}, n_{\circ AB\bullet}, n_{\bullet AB\bullet}\rangle)}{dt} = \\
&\quad k_{A+}(n_A+1)P_t(\langle n_A{+}1, n_{A^\star}{-}1, n_B, n_{B^\star}, n_{\circ AB\circ}, n_{\bullet AB\circ}, n_{\circ AB\bullet}, n_{\bullet AB\bullet}\rangle) \\
&+ k_{A+}(n_{\circ AB\circ}+1)P_t(\langle n_A, n_{A^\star}, n_B, n_{B^\star}, n_{\circ AB\circ}{+}1, n_{\bullet AB\circ}{-}1, n_{\circ AB\bullet}, n_{\bullet AB\bullet}\rangle) \\
&+ k_{A+}(n_{\circ AB\bullet}+1)P_t(\langle n_A, n_{A^\star}, n_B, n_{B^\star}, n_{\circ AB\circ}, n_{\bullet AB\circ}, n_{\circ AB\bullet}{+}1, n_{\bullet AB\bullet}{-}1\rangle) \\
&+ k_{A-}(n_{A^\star}+1)P_t(\langle n_A{-}1, n_{A^\star}{+}1, n_B, n_{B^\star}, n_{\circ AB\circ}, n_{\bullet AB\circ}, n_{\circ AB\bullet}, n_{\bullet AB\bullet}\rangle) \\
&+ k_{A-}(n_{\bullet AB\circ}+1)P_t(\langle n_A, n_{A^\star}, n_B, n_{B^\star}, n_{\circ AB\circ}{-}1, n_{\bullet AB\circ}{+}1, n_{\circ AB\bullet}, n_{\bullet AB\bullet}\rangle) \\
&+ k_{A-}(n_{\bullet AB\bullet}+1)P_t(\langle n_A, n_{A^\star}, n_B, n_{B^\star}, n_{\circ AB\circ}, n_{\bullet AB\circ}, n_{\circ AB\bullet}{-}1, n_{\bullet AB\bullet}{+}1\rangle) \\
&+ k_{B+}(n_B+1)P_t(\langle n_A, n_{A^\star}, n_B{+}1, n_{B^\star}{-}1, n_{\circ AB\circ}, n_{\bullet AB\circ}, n_{\circ AB\bullet}, n_{\bullet AB\bullet}\rangle) \\
&+ k_{B+}(n_{\circ AB\circ}+1)P_t(\langle n_A, n_{A^\star}, n_B, n_{B^\star}, n_{\circ AB\circ}{+}1, n_{\bullet AB\circ}, n_{\circ AB\bullet}{-}1, n_{\bullet AB\bullet}\rangle) \\
&+ k_{B+}(n_{\bullet AB\circ}+1)P_t(\langle n_A, n_{A^\star}, n_B, n_{B^\star}, n_{\circ AB\circ}, n_{\bullet AB\circ}{+}1, n_{\circ AB\bullet}, n_{\bullet AB\bullet}{-}1\rangle) \\
&+ k_{B-}(n_{B^\star}+1)P_t(\langle n_A, n_{A^\star}, n_B{-}1, n_{B^\star}{+}1, n_{\circ AB\circ}, n_{\bullet AB\circ}, n_{\circ AB\bullet}, n_{\bullet AB\bullet}\rangle) \\
&+ k_{B-}(n_{\circ AB\bullet}+1)P_t(\langle n_A, n_{A^\star}, n_B, n_{B^\star}, n_{\circ AB\circ}{-}1, n_{\bullet AB\circ}, n_{\circ AB\bullet}{+}1, n_{\bullet AB\bullet}\rangle) \\
&+ k_{B-}(n_{\bullet AB\bullet}+1)P_t(\langle n_A, n_{A^\star}, n_B, n_{B^\star}, n_{\circ AB\circ}, n_{\bullet AB\circ}{-}1, n_{\circ AB\bullet}, n_{\bullet AB\bullet}{+}1\rangle) \\
&+ k_{AB}(n_A+1)(n_B+1)P_t(\langle n_A{+}1, n_{A^\star}, n_B{+}1, n_{B^\star}, n_{\circ AB\circ}{-}1, n_{\bullet AB\circ}, n_{\circ AB\bullet}, n_{\bullet AB\bullet}\rangle) \\
&+ k_{AB}(n_A+1)(n_{B^\star}+1)P_t(\langle n_A{+}1, n_{A^\star}, n_B, n_{B^\star}{+}1, n_{\circ AB\circ}, n_{\bullet AB\circ}, n_{\circ AB\bullet}{-}1, n_{\bullet AB\bullet}\rangle) \\
&+ k_{AB}((n_{A^\star}+1)(n_B+1))P_t(\langle n_A, n_{A^\star}{+}1, n_B{+}1, n_{B^\star}, n_{\circ AB\circ}, n_{\bullet AB\circ}{-}1, n_{\circ AB\bullet}, n_{\bullet AB\bullet}\rangle) \\
&+ k_{AB\star}((n_{A^\star}+1)(n_{B^\star}+1))P_t(\langle n_A, n_{A^\star}{+}1, n_B, n_{B^\star}{+}1, n_{\circ AB\circ}, n_{\bullet AB\circ}, n_{\circ AB\bullet}, n_{\bullet AB\bullet}{-}1\rangle) \\
&+ k_{A..B}(n_{\circ AB\circ}-1)P_t(\langle n_A{-}1, n_{A^\star}{-}1, n_B, n_{B^\star}, n_{\circ AB\circ}{+}1, n_{\bullet AB\circ}, n_{\circ AB\bullet}, n_{\bullet AB\bullet}\rangle) \\
&+ k_{A..B}(n_{\circ AB\bullet}-1)P_t(\langle n_A{-}1, n_{A^\star}, n_B, n_{B^\star}{-}1, n_{\circ AB\circ}, n_{\bullet AB\circ}, n_{\circ AB\bullet}{+}1, n_{\bullet AB\bullet}\rangle) \\
&+ k_{A..B}(n_{\bullet AB\circ}-1)P_t(\langle n_A, n_{A^\star}{-}1, n_B{-}1, n_{B^\star}, n_{\circ AB\circ}, n_{\bullet AB\circ}{+}1, n_{\circ AB\bullet}, n_{\bullet AB\bullet}\rangle) \\
&+ k_{A..B}(n_{\bullet AB\bullet}-1)P_t(\langle n_A, n_{A^\star}{-}1, n_B, n_{B^\star}{-}1, n_{\circ AB\circ}, n_{\bullet AB\circ}, n_{\circ AB\bullet}, n_{\bullet AB\bullet}{+}1\rangle) \\
&- k_{A+}(n_A+n_{\circ AB\circ}+n_{\circ AB\bullet})P_t(\langle n_A, n_{A^\star}, n_B, n_{B^\star}, n_{\circ AB\circ}, n_{\bullet AB\circ}, n_{\circ AB\bullet}, n_{\bullet AB\bullet}\rangle) \\
&- k_{A-}(n_{A^\star}+n_{\bullet AB\circ}+n_{\bullet AB\bullet})P_t(\langle n_A, n_{A^\star}, n_B, n_{B^\star}, n_{\circ AB\circ}, n_{\bullet AB\circ}, n_{\circ AB\bullet}, n_{\bullet AB\bullet}\rangle) \\
&- k_{B+}(n_B+n_{\circ AB\circ}+n_{\bullet AB\circ})P_t(\langle n_A, n_{A^\star}, n_B, n_{B^\star}, n_{\circ AB\circ}, n_{\bullet AB\circ}, n_{\circ AB\bullet}, n_{\bullet AB\bullet}\rangle) \\
&- k_{B-}(n_{B^\star}+n_{AB^\star}+n_{\bullet AB\bullet})P_t(\langle n_A, n_{A^\star}, n_B, n_{B^\star}, n_{\circ AB\circ}, n_{\bullet AB\circ}, n_{\circ AB\bullet}, n_{\bullet AB\bullet}\rangle) \\
&- k_{AB}((n_A+n_{A^\star})(n_B+n_{B^\star})-n_{A^\star}n_{B^\star})P_t(\langle n_A, n_{A^\star}, n_B, n_{B^\star}, n_{\circ AB\circ}, n_{\bullet AB\circ}, n_{\circ AB\bullet}, n_{\bullet AB\bullet}\rangle) \\
&- k_{AB\star}n_{A^\star}n_{B^\star}P_t(\langle n_A, n_{A^\star}, n_B, n_{B^\star}, n_{\circ AB\circ}, n_{\bullet AB\circ}, n_{\circ AB\bullet}, n_{\bullet AB\bullet}\rangle) \\
&- k_{A..B}(n_{\circ AB\circ}+n_{\circ AB\bullet}+n_{\bullet AB\circ}+n_{\bullet AB\bullet})P_t(\langle n_A, n_{A^\star}, n_B, n_{B^\star}, n_{\circ AB\circ}, n_{\bullet AB\circ}, n_{\circ AB\bullet}, n_{\bullet AB\bullet}\rangle)
\end{aligned}
$$

Figure 5.8: Équation maîtresse pour le modèle avec une dissociation inconditionnelle.

5.2.2.2 Réduction de l'équation maîtresse

Cependant, la corrélation entre l'état de phosphorylation des protéines A et B dans les configurations de dimères empêche de réduire la sémantique stochastique de cet exemple. C'est ce qui est expliqué ici.

Dans la sémantique stochastique, l'état du système est représenté par un vecteur composé de 8 entiers naturels, $\langle n_A, n_{A^\star}, n_B, n_{B^\star}, n_{\circ AB\circ}, n_{\bullet AB\circ}, n_{\circ AB\bullet}, n_{\bullet AB\bullet}\rangle$, où n_X désigne le nombre d'occurrences de la configuration d'espèces biochimiques X dans l'état du système, pour tout $X \in \{A, A^\star, B, B^\star, AB, \bullet AB\circ, \circ AB\bullet, \bullet AB\bullet\}$. La probabilité $P_t(\sigma)$ que le système soit dans un état donné σ au temps t est alors donné par l'équation maîtresse donnée en figure 5.8.

Comme dans l'exemple de la section 5.2.1, nous voudrions passer sous silence la corrélation entre les états de phosphorylation des occurrences des protéines A et B dans les configurations du dimère AB. Ceci revient à representer l'état du système $\sigma = \langle n_A, n_{A^\star}, n_B, n_{B^\star}, n_{\circ AB\circ}, n_{\bullet AB\circ}, n_{\circ AB\bullet}, n_{\bullet AB\bullet}\rangle$ par le vecteur de 8 entiers naturels $\beta(\sigma) = \langle n_A, n_{A^\star}, n_B, n_{B^\star}, n_{\circ AB\circ}+n_{\circ AB\bullet}, n_{\bullet AB\circ}+n_{\bullet AB\bullet}, n_{\circ AB\circ}+n_{\bullet AB\circ}, n_{\circ AB\bullet}+n_{\bullet AB\bullet}\rangle$. Le vecteur $\beta(\sigma)$ sera alors noté $\langle n_{\circ A\dashv}, n_{\bullet A\dashv}, n_{\dashv B\circ}, n_{\dashv B\bullet}, n_{\circ A-}, n_{\bullet A-}, n_{-B\circ}, n_{-B\bullet}\rangle$ en s'inspirant de la syntaxe du langage Kappa. La probabilité $P_t^\sharp(\sigma^\sharp)$ que le système soit dans un état σ tel que $\beta(\sigma) = \sigma^\sharp$ à l'instant t est définie par l'équation donnée en figure 5.9. L'enjeu de cette équation est d'exprimer les propensités des différentes réactions à partir uniquement des composantes de l'état réduit $\sigma^\sharp$. Cela pose des difficultés pour les réactions de dissociation. Il faut en effet être capable de retrouver la distribution du nombre d'occurrences des différentes configurations du dimère, ne connaissant que le nombre d'occurrences de la configuration de la protéine A liée et non phosphorylée, le nombre d'occurrences de celle de la protéine A liée et phosphorylée, le nombre d'occurrences de celle de la protéine B liée et non phosphorylée et le nombre d'occurrences de celle de la protéine B liée et phosphorylée. En fait, il suffit de connaître l'espérance du nombre d'occurrences des différentes configurations du dimère sachant que la réduction de l'état du système est égale à $\sigma^\sharp$. Ces espérances conditionnelles peuvent

$$
\begin{aligned}
&\frac{dP_t^\sharp(\langle n_{\circ A\dashv}, n_{\bullet A\dashv}, n_{\vdash B\circ}, n_{\vdash B\bullet}, n_{\circ A-}, n_{\bullet A-}, n_{-B\circ}, n_{-B\bullet}\rangle)}{dt} = \\
&\quad k_{A+}(n_A+1)P_t^\sharp(\langle n_{\circ A\dashv}+1, n_{\bullet A\dashv}-1, n_{\vdash B\circ}, n_{\vdash B\bullet}, n_{\circ A-}, n_{\bullet A-}, n_{-B\circ}, n_{-B\bullet}\rangle) \\
&+ k_{A+}(n_{\circ A-}+1)P_t^\sharp(\langle n_{\circ A\dashv}, n_{\bullet A\dashv}, n_{\vdash B\circ}, n_{\vdash B\bullet}, n_{\circ A-}+1, n_{\bullet A-}-1, n_{-B\circ}, n_{-B\bullet}\rangle) \\
&+ k_{A-}(n_{A^\star}+1)P_t^\sharp(\langle n_{\circ A\dashv}-1, n_{\bullet A\dashv}+1, n_{\vdash B\circ}, n_{\vdash B\bullet}, n_{\circ A-}, n_{\bullet A-}, n_{-B\circ}, n_{-B\bullet}\rangle) \\
&+ k_{A-}(n_{\bullet A-}+1)P_t^\sharp(\langle n_{\circ A\dashv}, n_{\bullet A\dashv}, n_{\vdash B\circ}, n_{\vdash B\bullet}, n_{\circ A-}-1, n_{\bullet A-}+1, n_{-B\circ}, n_{-B\bullet}\rangle) \\
&+ k_{B+}(n_B+1)P_t^\sharp(\langle n_{\circ A\dashv}, n_{\bullet A\dashv}, n_{\vdash B\circ}+1, n_{\vdash B\bullet}-1, n_{\circ A-}, n_{\bullet A-}, n_{-B\circ}, n_{-B\bullet}\rangle) \\
&+ k_{B+}(n_{-B\circ}+1)P_t^\sharp(\langle n_{\circ A\dashv}, n_{\bullet A\dashv}, n_{\vdash B\circ}, n_{\vdash B\bullet}, n_{\circ A-}, n_{\bullet A-}, n_{-B\circ}+1, n_{-B\bullet}-1\rangle) \\
&+ k_{B-}(n_{B^\star}+1)P_t^\sharp(\langle n_{\circ A\dashv}, n_{\bullet A\dashv}, n_{\vdash B\circ}-1, n_{\vdash B\bullet}+1, n_{\circ A-}, n_{\bullet A-}, n_{-B\circ}, n_{-B\bullet}\rangle) \\
&+ k_{B-}(n_{-B\bullet}+1)P_t^\sharp(\langle n_{\circ A\dashv}, n_{\bullet A\dashv}, n_{\vdash B\circ}, n_{\vdash B\bullet}, n_{\circ A-}, n_{\bullet A-}, n_{-B\circ}-1, n_{-B\bullet}+1\rangle) \\
&+ k_{AB}(n_A+1)(n_B+1)P_t^\sharp(\langle n_{\circ A\dashv}+1, n_{\bullet A\dashv}, n_{\vdash B\circ}+1, n_{\vdash B\bullet}, n_{\circ A-}-1, n_{\bullet A-}, n_{-B\circ}-1, n_{-B\bullet}\rangle) \\
&+ k_{AB}(n_A+1)(n_{B^\star}+1)P_t^\sharp(\langle n_{\circ A\dashv}+1, n_{\bullet A\dashv}, n_{\vdash B\circ}, n_{\vdash B\bullet}+1, n_{\circ A-}-1, n_{\bullet A-}, n_{-B\circ}, n_{-B\bullet}-1\rangle) \\
&+ k_{AB}((n_{A^\star}+1)(n_B+1))P_t^\sharp(\langle n_{\circ A\dashv}, n_{\bullet A\dashv}+1, n_{\vdash B\circ}+1, n_{\vdash B\bullet}, n_{\circ A-}, n_{\bullet A-}-1, n_{-B\circ}-1, n_{-B\bullet}\rangle) \\
&+ k_{AB\star}((n_{A^\star}+1)(n_{B^\star}+1))P_t^\sharp(\langle n_{\circ A\dashv}, n_{\bullet A\dashv}+1, n_{\vdash B\circ}, n_{\vdash B\bullet}+1, n_{\circ A-}, n_{\bullet A-}-1, n_{-B\circ}, n_{-B\bullet}-1\rangle) \\
&+ k_{A..B}\tilde{E}_t(n_{\circ AB\circ} \mid \langle n_{\circ A\dashv}-1, n_{\bullet A\dashv}, n_{\vdash B\circ}-1, n_{\vdash B\bullet}, n_{\circ A-}+1, n_{\bullet A-}, n_{-B\circ}+1, n_{-B\bullet}\rangle) \\
&+ k_{A..B}\tilde{E}_t(n_{\circ AB\bullet} \mid \langle n_A-1, n_{A^\star}, n_B, n_{B^\star}-1, n_{\circ AB\circ}+1, n_{\bullet AB\circ}, n_{\circ AB\bullet}, n_{\bullet AB\bullet}+1\rangle) \\
&+ k_{A..B}\tilde{E}_t(n_{\bullet AB\circ} \mid \langle n_A, n_{A^\star}-1, n_B-1, n_{B^\star}, n_{\circ AB\circ}, n_{\bullet AB\circ}+1, n_{\circ AB\bullet}+1, n_{\bullet AB\bullet}\rangle) \\
&+ k_{A..B}\tilde{E}_t(n_{\bullet AB\bullet} \mid \langle n_A, n_{A^\star}-1, n_B, n_{B^\star}-1, n_{\circ AB\circ}, n_{\bullet AB\circ}+1, n_{\circ AB\bullet}, n_{\bullet AB\bullet}+1\rangle) \\
&- k_{A+}(n_A+n_{\circ A-})P_t^\sharp(\langle n_{\circ A\dashv}, n_{\bullet A\dashv}, n_{\vdash B\circ}, n_{\vdash B\bullet}, n_{\circ A-}, n_{\bullet A-}, n_{-B\circ}, n_{-B\bullet}\rangle) \\
&- k_{A-}(n_{A^\star}+n_{\bullet A-})P_t^\sharp(\langle n_{\circ A\dashv}, n_{\bullet A\dashv}, n_{\vdash B\circ}, n_{\vdash B\bullet}, n_{\circ A-}, n_{\bullet A-}, n_{-B\circ}, n_{-B\bullet}\rangle) \\
&- k_{B+}(n_B+n_{-B\circ})P_t^\sharp(\langle n_{\circ A\dashv}, n_{\bullet A\dashv}, n_{\vdash B\circ}, n_{\vdash B\bullet}, n_{\circ A-}, n_{\bullet A-}, n_{-B\circ}, n_{-B\bullet}\rangle) \\
&- k_{B-}(n_{B^\star}+n_{-B\bullet})P_t^\sharp(\langle n_{\circ A\dashv}, n_{\bullet A\dashv}, n_{\vdash B\circ}, n_{\vdash B\bullet}, n_{\circ A-}, n_{\bullet A-}, n_{-B\circ}, n_{-B\bullet}\rangle) \\
&- k_{AB}((n_A+n_{A^\star})(n_B+n_{B^\star})-n_{A^\star}n_{B^\star})P_t^\sharp(\langle n_{\circ A\dashv}, n_{\bullet A\dashv}, n_{\vdash B\circ}, n_{\vdash B\bullet}, n_{\circ A-}, n_{\bullet A-}, n_{-B\circ}, n_{-B\bullet}\rangle) \\
&- k_{AB\star}(n_{A^\star}n_{B^\star})P_t^\sharp(\langle n_{\circ A\dashv}, n_{\bullet A\dashv}, n_{\vdash B\circ}, n_{\vdash B\bullet}, n_{\circ A-}, n_{\bullet A-}, n_{-B\circ}, n_{-B\bullet}\rangle) \\
&- k_{A..B}(n_{\bullet A-}+n_{\circ A-})P_t^\sharp(\langle n_{\circ A\dashv}, n_{\bullet A\dashv}, n_{\vdash B\circ}, n_{\vdash B\bullet}, n_{\circ A-}, n_{\bullet A-}, n_{-B\circ}, n_{-B\bullet}\rangle),
\end{aligned}
$$

Figure 5.9: Tentative de réduction de l'équation maîtresse pour l'exemple avec la dissociation inconditionnelle. Pour toute expression $X(\sigma)$ et tout état réduit $\sigma^\sharp$, l'expression $\tilde{E}_t(X(\sigma) \mid \sigma^\sharp)$ représente le produit entre, d'une part, l'espérance conditionnelle $E_t(X(\sigma) \mid \sigma^\sharp)$ de l'expression $X(\sigma)$ sachant que l'état du système satisfait la contrainte $\beta(\sigma) = \sigma^\sharp$ et, d'autre part, la probabilité $P_t^\sharp(\sigma^\sharp)$ d'être dans un état σ tel que $\beta(\sigma) = \sigma^\sharp$. Les nombres d'occurrences des différentes configurations du dimère n'étant pas exprimable à partir de $\sigma^\sharp$, ceux-ci sont remplacés par leurs espérances conditionnelles sachant que l'état du système σ vérifie la condition $\beta(\sigma) = \sigma^\sharp$. La propensité obtenue est alors multipliée par la probabilité d'être dans un état σ tel que $\beta(\sigma) = \sigma^\sharp$ avant d'effectuer cette interaction.

être calculées à partir de l'équation maîtresse initiale. *A priori*, ces espérances dépendent du temps et ne peuvent pas être exprimées uniquement à partir des informations présentes dans la réduction $\sigma^\sharp$ de l'état du système. Chaque propensité étant multipliée par la probabilité d'être dans l'état source à l'instant t en question, la notation $\tilde{E}_t(X(\sigma) \mid \sigma^\sharp)$ est introduite pour représenter pour toute expression $X(\sigma)$ et tout état réduit $\sigma^\sharp$, le produit entre, d'une part, l'espérance conditionnelle $E_t(X(\sigma) \mid \sigma^\sharp)$ de l'expression $X(\sigma)$ sachant que le système est dans un état σ tel que la contrainte $\beta(\sigma) = \sigma^\sharp$ soit satisfaite et, d'autre part, la probabilité $P_t^\sharp(\sigma^\sharp)$ d'être dans un état σ tel que $\beta(\sigma) = \sigma^\sharp$.

Il y a alors deux cas selon que les constantes d'interactions k_{AB} et $k_{AB\star}$ soient égales ou non.

1. Lorsque $k_{AB} = k_{AB\star}$, il est possible de vérifier que les probabilités $P_t(\sigma)$ et $P_t(\sigma')$ sont égales à tout instant, pour toute paire d'états σ et σ' telle que $\beta(\sigma) = \beta(\sigma')$, dès lors que $P_0(\sigma)$ et $P_0(\sigma')$ pour toute paire d'états σ et σ' telle que $\beta(\sigma) = \beta(\sigma')$ dans la distribution à l'intant initial. Aussi, à la condition que $k_{AB} = k_{AB\star}$ et qu'il n'y a pas de corrélation entre les états de phosphorylation des occurrences des protéines A et B dans les configurations du dimère AB à l'instant $t = 0$, les quatre équations suivantes sont satisfaites:

$$E_t(n_{\circ AB\circ} \mid \langle n_{\circ A+}, n_{\bullet A+}, n_{+B\circ}, n_{+B\bullet}, n_{\circ A-}, n_{\bullet A-}, n_{-B\circ}, n_{-B\bullet}\rangle) = \frac{n_{\circ A-} \cdot n_{-B\circ}}{n_{\circ A-} + n_{\bullet A-}}$$

$$E_t(n_{\circ AB\bullet} \mid \langle n_{\circ A+}, n_{\bullet A+}, n_{+B\circ}, n_{+B\bullet}, n_{\circ A-}, n_{\bullet A-}, n_{-B\circ}, n_{-B\bullet}\rangle) = \frac{n_{\circ A-} \cdot n_{-B\bullet}}{n_{\circ A-} + n_{\bullet A-}}$$

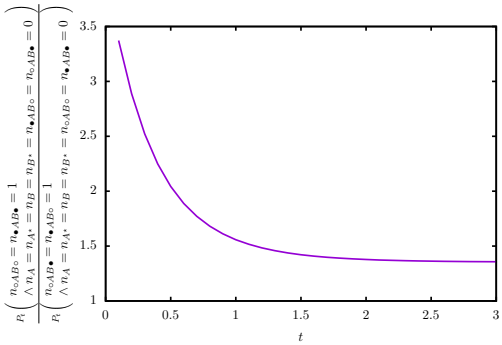
$$E_t(n_{\bullet AB\circ} \mid \langle n_{\circ A+}, n_{\bullet A+}, n_{+B\circ}, n_{+B\bullet}, n_{\circ A-}, n_{\bullet A-}, n_{-B\circ}, n_{-B\bullet}\rangle) = \frac{n_{\bullet A-} \cdot n_{-B\circ}}{n_{\circ A-} + n_{\bullet A-}}$$

$$E_t(n_{\bullet AB\bullet} \mid \langle n_{\circ A+}, n_{\bullet A+}, n_{+B\circ}, n_{+B\bullet}, n_{\circ A-}, n_{\bullet A-}, n_{-B\circ}, n_{-B\bullet}\rangle) = \frac{n_{\bullet A-} \cdot n_{-B\bullet}}{n_{\circ A-} + n_{\bullet A-}}.$$

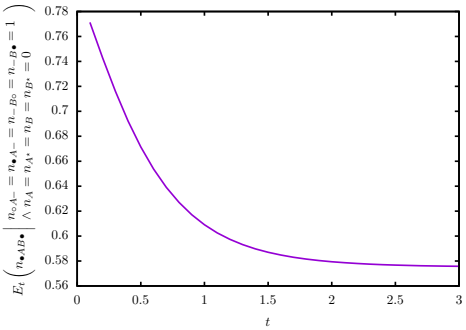
   Ceci permet d'exprimer les espérances conditionnelles des expressions $n_{\circ AB\circ}$, $n_{\circ AB\bullet}$, $n_{\bullet AB\circ}$ et $n_{\bullet AB\bullet}$ uniquement en fonction des expressions $n_{\circ A-}$, $n_{\bullet A-}$, $n_{-B\circ}$ et $n_{-B\bullet}$.

   Dans ce cas, il est donc possible de réduire l'équation maîtresse du système.

2. Lorsque $k_{AB} \neq k_{AB\star}$, les espérances conditionnelles qui apparaissent dans la tentative de réduction de l'équation maîtresse peuvent varier en fonction du temps. Ainsi en figure 5.10(b) est montré l'évolution temporelle de l'espérance conditionnelle du nombre d'occurrences de la configuration doublement phosphorylée du dimère sachant que l'état du système contient une occurrence de la configuration de la protéine A liée et phosphorylée, une occurrence de la configuration de la protéine A liée et non phosphorylée, une occurrence de la configuration de la protéine B liée et phosphorylée et une occurrence de la configuration de la protéine B liée et non phosphorylée. Cette courbe est obtenue en intégrant numériquement l'équation maîtresse initiale du modèle en prenant pour distribution d'états initiale, l'unique état formé de deux occurrences de la configuration de la protéine A libre et non phosphorylée et deux occurrences de la configuration de la protéine B libre et non phosphorylée. Les taux d'interactions k_{A+}, k_{B+}, k_{A-}, k_{B-}, k_{AB} et $k_{A..B}$ sont fixées à 1, alors que la constante $k_{AB\star}$ est fixée à 10 (ce qui modélise bien qu'une liaison entre deux occurrences de configurations phosphorylées des protéines A et B est plus facile que les autres). Cette espérance conditionnelle varie avec le temps et on ne sait pas l'exprimer en fonction des probabilités d'être dans un σ tel que $\beta(\sigma) = \sigma^\sharp$, pour tout état réduit $\sigma^\sharp$. En particulier, il serait faux de considérer que les états de phosphorylation des occurrences de configurations des protéines A et B dans les occurrences des configurations de dimère sont indépendants. En effet, en figure 5.10(a) est dessiné comment évolue le quotient entre la probabilité que le système contienne une occurrence de la configuration entièrement non phosphorylée du dimère et une occurrence de la configuration entièrement phosphorylée du dimère et celle qu'il contienne une occurrence de la configuration du dimère avec une seule phosphorylation portée par l'occurrence de la protéine A et une occurrence de la configuration du dimère avec une seule phosphorylation portée par l'occurrence de la protéine B. Les conditions prises pour la distribution de l'état initiale et les taux d'interaction sont les mêmes que pour la courbe de la figure 5.10(b). Si l'état des configurations des protéines A et B étaient indépendants, alors ces deux probabilités seraient toujours égales. Ce n'est pas la cas, ce qui montre l'existence d'une corrélation entre l'état de phosphorylations des occurrences des configurations de la protéine A et B dans les occurrences des configurations du dimère. Il est à remarquer que l'espérance conditionnelle du nombre d'occurrences de configuration du dimère doublement phosphorylée tends vers une constante. Il ne faut pas y attribuer une



(a) Évolution temporelle du quotient de la probabilité que le système soit dans un état formé d'une occurrence de la configuration du dimère entièrement non phosphorylée et d'une autre de celle du dimère entièrement phosphorylée et de la probabilité que le système soit dans un état formé d'une occurrence de la configuration du dimère phosphorylée sur l'occurrence de la protéine A, mais pas celle de la protéine B, et d'une autre de celle du dimère phosphorylée sur l'occurrence de la protéine B, mais pas celle de la protéine A.



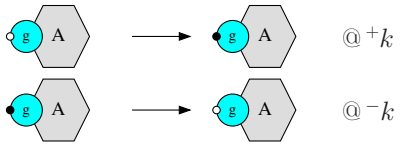
(b) Évolution temporelle de l'espérance conditionnelle du nombre d'occurrences de la configuration du dimère doublement phosphorylée sachant que l'état contient exactement deux occurrences du dimère et aucune occurrence de protéines isolées.

Figure 5.10: Évolution de quelques propriétés statistiques du modèle avec une liaison inconditionnelle. La distribution initiale est réduite à un état contenant uniquement quatre occurrences de protéines, à savoir deux occurrences de la configuration de la protéine A, libre et non phosphorylée, et deux occurrences de la configuration de la protéine B, libre et non phosphorylée. Toutes les constantes d'interaction sont fixées à 1, sauf la constante $k_{AB\star}$ qui est fixé à 10. En 5.10(a) sont comparées la probabilité que le système contiennent deux occurrences du dimère en configuration soit sans aucune phosphorylation, soit totalement phosphorylée et la probabilité que le système contiennent deux occurrences du dimère en configuration avec exactement une des deux occurrences de protéines phosphorylée. Si les état de phosphorylation des occurrences de la protéine A et celui de celles de la protéine B étaient indépendants dans les occurrences du dimère, ces deux probabilités seraient égales. Ce n'est visiblement pas le cas. En 5.10(b) est représentée l'évolution temporelle de l'espérance conditionnelle du nombre d'occurrences de la configuration doublement phosphorylée du dimère, sachant que les 4 occurrences de protéines de l'état initial forment des dimères et qu'exactement une occurrence de chaque protéine est phosphorylée.

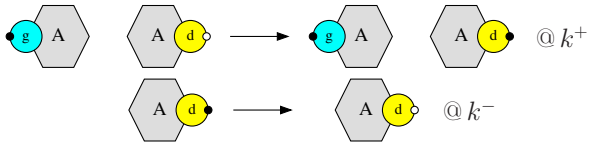
interprétation particulière. Cela vient juste du fait que la solution de l'équation maîtresse converge vers une distribution d'états stationnaire.

5.2.2.3 Conclusion

Ainsi dans cette exemple, il n'est pas possible, dans la sémantique stochastique, de faire abstraction de la corrélation entre les configurations prises par les deux occurrences de protéines d'un dimère. En effet, chaque application de la règle de dissociation transforme simultanément les configurations des deux occurrences de protéines qui sont liées. Or le choix de ces deux configurations est justement contraint par cette corrélation. Cette difficulté ne se rencontre pas dans le cadre différentiel. Dans ce dernier, l'action de dissociation s'applique simultanément sur une partie équitable des quantités de chaque configuration possible pour les occurrences de protéines, ce qui permet de simplifier le modèle. Un phénomène analogue se produit dans le cas où les fragments qui sont utilisés pour réduire la sémantique différentielle d'un modèle se chevauchent. Dans le cadre stochastique, pour appliquer une action sur la partie commune à deux fragments, il faut choisir simultanément une configuration pour chacun de ces deux fragments. Or ce choix serait contraint par la corrélation entre les différentes configurations qu'ils peuvent prendre. Ainsi dans l'exemple 4.2 page 45 , si une règle pour déphosphoryler le site c sans condition est ajoutée, il est toujours possible de réduire la sémantique différentielle, mais cette simplification devient impossible pour la sémantique stochastique.



(a) Phosphorylation/déphosphorylation du site de gauche.



(b) Phosphorylation/déphosphorylation du site de droite.

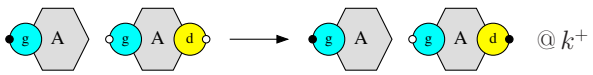
Figure 5.11: Les règles d'interaction pour le modèle avec la phosphorylation à distance. En 5.11(a), le site gauche de la protéine peut se phosphoryler et se déphosphoryler sans conditions. En 5.11(b), le taux de phosphorylation du site droit est proportionnel à la quantité de configurations de la protéine dont le site gauche est phosphrylé. La déphosphorylation du site droit est, elle, inconditionnelle.

5.2.3 Un exemple de contrôle à distance

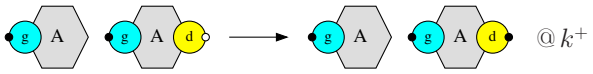
Dans ce dernier exemple, une protéine peut exercer un contrôle sur le comportement d'une autre occurrence de cette même protéine même s'ils ne sont pas dans la même composante connexe dans le membre gauche de la règle en question. Si cela n'empêche pas la réduction du système différentiel engendré par ces règles, il sera expliqué pourquoi on ne sait pas en simplifier la sémantique stochastique.

Soit A une protéine qui comporte deux sites de phosphorylation, g et d. Le site g sera représenté à gauche des occurrences des configurations de la protéine et le site d à droite. Chacun de ces sites peut être phosphorylé ou non. Les règles de phosphorylation et de déphosphorylation sont données en figure 5.11. En particulier, la cinétique de la phosphorylation du site de droite de la protéine dépends de la quantité de configurations de la protéine avec le site gauche déjà phosphorylé (voir la première règle de la figure 5.11(b)) : cela signifie que les occurrences des configurations de la protéine avec le site gauche phosphorylé catalysent la phosphorylation du site droit des autres occurrences des configurations de la protéine. Les autres règles sont, elles, purement locales. Ainsi, en figure 5.11(a), la phosphorylation et la déphosphorylation du site de gauche ne dépendent ni de l'état de phosphorylation du site de droite, ni de la quantité de configurations de la protéine déjà phosphorylée sur un ou deux sites. De la même manière, sur la seconde règle de la figure 5.11(b), la déphosphorylation du site de droite ne dépend ni de l'état de phosphorylation du site de gauche, ni de la quantité de configurations de la protéine déjà phosphorylée sur un ou deux sites.

Dans cet exemple, le but est de voir s'il est possible de faire abstraction de la corrélation entre l'état de phosphorylation des deux sites dans les occurrences des configurations de la protéine. Cela revient à découper la protéine en deux parties — gauche et droite — tout en oubliant quelles occurrences de configurations des moitiés de la protéine correspondent à des occurrences de configurations de la protéine entière. Cela pose un problème pour simplifier la sémantique stochastique. En effet, la règle d'activation du site droit de la protéine favorise la phosphorylation des occurrences de la protéine dans la configuration entièrement non phosphorylée. Par exemple, étant données m occurrences de la protéine dans la configuration entièrement non phosphorylée, m occurrences de la protéine dans la configuration avec le site gauche phosphorylé mais pas le site droit et n occurrences de la protéine dans la configuration doublement phosphorylée (le nombre d'occurrences de la protéine dans la configuration avec le site droit phosphorylé, mais pas le gauche n'a pas d'importance), le nombre d'applications potentielles du raffinement de la règle suivant :



est égal à $m \cdot (n + m)$, alors que celui du raffinement suivant :



est égal à $m \cdot (n + m - 1)m$ (la soustraction par 1 vient du fait que les deux occurrences de configurations de la protéine du membre gauche d'une règle doivent être associées à des occurrences différentes de la protéine). Cependant, ceci n'empêche pas de séparer les configurations de la protéine en deux et d'oublier la corrélation entre leurs nombres d'occurrences dans la sémantique différentielle. Intuitivement, le terme 1 disparaît devant une infinité d'occurrences de configurations de la protéine (dans un volume également infini).

5.2.3.1 Réduction de la sémantique différentielle

Vérifions cela formellement. La sémantique différentielle du modèle est définie par le système d'équations différentielles suivant :

$$\begin{aligned}
\frac{d\,[\mathrm{A}]}{dt} &= {}^{-}k\,[\star\mathrm{A}] + k^{-}\,[\mathrm{A}\star] - ({}^{+}k + k^{+}([\star\mathrm{A}] + [\star\mathrm{A}\star]))\,[\mathrm{A}] \\
\frac{d\,[\star\mathrm{A}]}{dt} &= {}^{+}k\,[\mathrm{A}] + k^{-}\,[\star\mathrm{A}\star] - ({}^{-}k + k^{+}([\star\mathrm{A}] + [\star\mathrm{A}\star]))\,[\star\mathrm{A}] \\
\frac{d\,[\mathrm{A}\star]}{dt} &= {}^{-}k\,[\star\mathrm{A}\star] + k^{+}\,[\mathrm{A}]\,([\star\mathrm{A}] + [\star\mathrm{A}\star]) - ({}^{+}k + k^{-})\,[\mathrm{A}\star] \\
\frac{d\,[\star\mathrm{A}\star]}{dt} &= {}^{+}k\,[\mathrm{A}\star] + k^{+}\,[\star\mathrm{A}]\,([\star\mathrm{A}] + [\star\mathrm{A}\star]) - ({}^{-}k + k^{-})\,[\star\mathrm{A}\star]\,.
\end{aligned}$$

Dans ce système, A désigne la configuration de la protéine sans phosphorylation, $\star$A celle avec le site gauche phosphorylé, mais pas le site droit, A$\star$ celle avec le site droit phosphorylé, mais pas le site gauche et $\star$A$\star$ celle avec les deux sites liés.

Il est possible d'ignorer la corrélation entre l'état de phosphorylation des deux sites de la protéine. En effet, les équations suivantes :

$$\begin{aligned}
\frac{d\,[\mathrm{A}\diamond]}{dt} &= {}^{-}k\,[\star\mathrm{A}\diamond] - {}^{+}k\,[\mathrm{A}\diamond] \\
\frac{d\,[\star\mathrm{A}\diamond]}{dt} &= {}^{+}k\,[\mathrm{A}\diamond] - {}^{-}k\,[\star\mathrm{A}\diamond] \\
\frac{d\,[\diamond\mathrm{A}]}{dt} &= k^{-}\,[\diamond\mathrm{A}\star] - k^{+}\,[\diamond\mathrm{A}]\,[\star\mathrm{A}\diamond] \\
\frac{d\,[\diamond\mathrm{A}\star]}{dt} &= k^{+}\,[\diamond\mathrm{A}]\,[\star\mathrm{A}\diamond] - k^{-}\,[\diamond\mathrm{A}\star]\,,
\end{aligned}$$

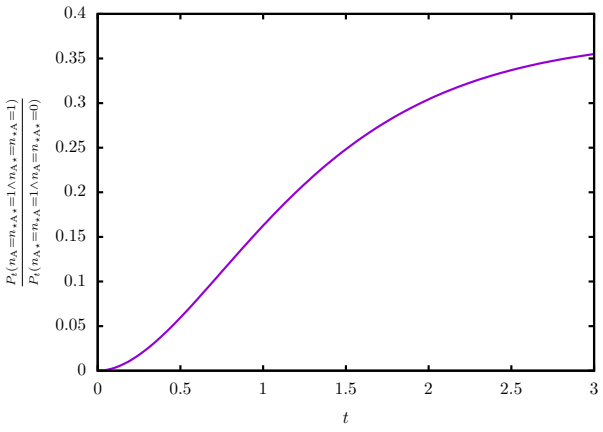
dans lesquelles $[\mathrm{A}\diamond] := [\mathrm{A}] + [\mathrm{A}\star]$, $[\star\mathrm{A}\diamond] := [\star\mathrm{A}] + [\star\mathrm{A}\star]$, $[\diamond\mathrm{A}] := [\mathrm{A}] + [\star\mathrm{A}]$ et $[\diamond\mathrm{A}\star] := [\mathrm{A}\star] + [\star\mathrm{A}\star]$, sont également satisfaites. Le symbole diamant représente le fait que l'état d'un site est passé sous silence, le site de gauche quand le diamant est à gauche de la lettre A et le site de droit quand le diamant est à droite de celle-ci.

5.2.3.2 Réduction de l'équation maîtresse

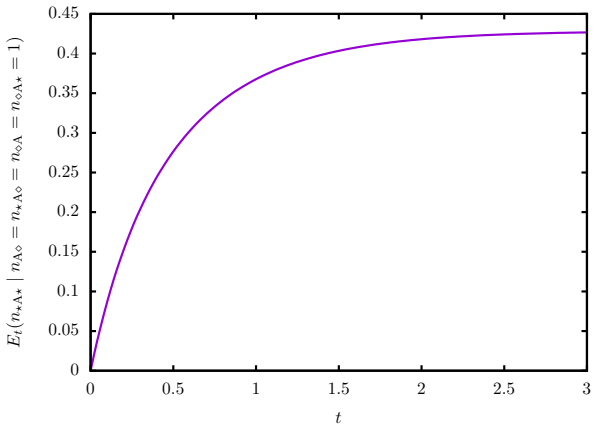
Est-ce possible de réduire la sémantique stochastique en procédant de la même manière ? L'équation maîtresse du modèle peut être examinée, pour répondre à cette question. L'état discret du système est représenté par un quadruplet $\langle n_{\mathrm{A}}, n_{\star\mathrm{A}}, n_{\mathrm{A}\star}, n_{\star\mathrm{A}\star}\rangle$ d'entier naturel, où n_X représente le nombre d'occurrences de la protéine dans la configuration X, pour toute configuration de la protéine dans l'ensemble $\{\mathrm{A}, \mathrm{A}\star, \star\mathrm{A}, \star\mathrm{A}\star\}$. La probabilité $P_t(\langle n_{\mathrm{A}}, n_{\star\mathrm{A}}, n_{\mathrm{A}\star}, n_{\star\mathrm{A}\star}\rangle)$ que le système soit dans l'état $\langle n_{\mathrm{A}}, n_{\star\mathrm{A}}, n_{\mathrm{A}\star}, n_{\star\mathrm{A}\star}\rangle$ à l'instant t est donnée par l'équation maîtresse suivante :

$$\begin{aligned}
&\frac{dP_t(\langle n_{\mathrm{A}}, n_{\star\mathrm{A}}, n_{\mathrm{A}\star}, n_{\star\mathrm{A}\star}\rangle)}{dt} = \\
&\quad {}^{+}k(n_{\mathrm{A}}+1)P_t(\langle n_{\mathrm{A}}{+}1, n_{\star\mathrm{A}}{-}1, n_{\mathrm{A}\star}, n_{\star\mathrm{A}\star}\rangle) \\
+\;& {}^{+}k(n_{\mathrm{A}\star}+1)P_t(\langle n_{\mathrm{A}}, n_{\star\mathrm{A}}, n_{\mathrm{A}\star}{+}1, n_{\star\mathrm{A}\star}{-}1\rangle) \\
+\;& {}^{-}k(n_{\star\mathrm{A}}+1)P_t(\langle n_{\mathrm{A}}{-}1, n_{\star\mathrm{A}}{+}1, n_{\mathrm{A}\star}, n_{\star\mathrm{A}\star}\rangle) \\
+\;& {}^{-}k(n_{\star\mathrm{A}\star}+1)P_t(\langle n_{\mathrm{A}}, n_{\star\mathrm{A}}, n_{\mathrm{A}\star}{-}1, n_{\star\mathrm{A}\star}{+}1\rangle) \\
+\;& k^{+}(n_{\mathrm{A}}+1)(n_{\star\mathrm{A}}+n_{\star\mathrm{A}\star})P_t(\langle n_{\mathrm{A}}{+}1, n_{\star\mathrm{A}}, n_{\mathrm{A}\star}{-}1, n_{\star\mathrm{A}\star}\rangle) \\
+\;& k^{+}(n_{\star\mathrm{A}}+1)(n_{\star\mathrm{A}}+n_{\star\mathrm{A}\star}-1)P_t(\langle n_{\mathrm{A}}, n_{\star\mathrm{A}}{+}1, n_{\mathrm{A}\star}, n_{\star\mathrm{A}\star}{-}1\rangle) \\
+\;& k^{-}(n_{\mathrm{A}\star}+1)P_t(\langle n_{\mathrm{A}}{-}1, n_{\star\mathrm{A}}, n_{\mathrm{A}\star}{+}1, n_{\star\mathrm{A}\star}\rangle) \\
+\;& k^{-}(n_{\star\mathrm{A}\star}+1)P_t(\langle n_{\mathrm{A}}, n_{\star\mathrm{A}}{-}1, n_{\mathrm{A}\star}, n_{\star\mathrm{A}\star}{+}1\rangle) \\
-\;& {}^{+}k(n_{\mathrm{A}}+n_{\mathrm{A}\star})P_t(\langle n_{\mathrm{A}}, n_{\star\mathrm{A}}, n_{\mathrm{A}\star}, n_{\star\mathrm{A}\star}\rangle) \\
-\;& {}^{-}k(n_{\star\mathrm{A}}+n_{\star\mathrm{A}\star})P_t(\langle n_{\mathrm{A}}, n_{\star\mathrm{A}}, n_{\mathrm{A}\star}, n_{\star\mathrm{A}\star}\rangle) \\
-\;& k^{+}(n_{\mathrm{A}}(n_{\star\mathrm{A}}+n_{\star\mathrm{A}\star})+n_{\star\mathrm{A}}(n_{\star\mathrm{A}}+n_{\star\mathrm{A}\star}-1))P_t(\langle n_{\mathrm{A}}, n_{\star\mathrm{A}}, n_{\mathrm{A}\star}, n_{\star\mathrm{A}\star}\rangle) \\
-\;& k^{-}(n_{\mathrm{A}\star}+n_{\star\mathrm{A}\star})P_t(\langle n_{\mathrm{A}}, n_{\star\mathrm{A}}, n_{\mathrm{A}\star}, n_{\star\mathrm{A}\star}\rangle).
\end{aligned}$$

Passer sous silence la corrélation entre les états de phosphorylation des deux sites de la protéine A dans les occurrences de ses configurations revient à representer l'état du système $\sigma = \langle n_{\mathrm{A}}, n_{\star\mathrm{A}}, n_{\mathrm{A}\star}, n_{\star\mathrm{A}\star}\rangle$ par un



(a) Évolution temporelle du quotient de la probabilité que le système soit dans un état formé d'une occurrence de la configuration de la protéine entièrement non phosphorylée et d'une autre de celle de la protéine doublement phosphorylée et de la probabilité que le système soit dans un état formé d'une occurrence de la configuration de la protéine phosphorylée uniquement à droite et d'une autre de celle de la protéine phosphorylée uniquement à gauche.



(b) Évolution temporelle de l'espérance conditionnelle du nombre d'occurrences de la configuration de la protéine doublement phosphorylée sachant que l'état contient exactement deux occurrences de configurations de la protéine, que le site gauche de la protéine est phosphorylé dans une seule de ces occurrences de configurations et que le site droit de la protéine est phosphorylé, lui-aussi, dans une seule (la même ou non) de ces occurrences de configurations.

Figure 5.12: Évolution de quelques propriétés statistiques du modèle avec un contrôle distant. La distribution initiale est réduite à un état contenant uniquement deux occurrences de protéines, dans la configuration entièrement non phosphorylée. Toutes les constantes d'interaction sont fixées à 1. En 5.12(a) sont comparées la probabilité que le système contiennent deux occurrences de la protéine, l'une dans sa configuration sans aucune phosphorylation et l'autre totalement phosphorylée et la probabilité que le système contiennent deux occurrences de la protéine l'une dans sa configuration phosphorylée à gauche, mais pas à droite et l'autre dans sa configuration phosphorylée à droite, mais pas à gauche. Si les états de phosphorylation des deux sites dans les configurations des occurrences de la protéine A étaient indépendants, ces deux probabilités seraient égales. Ce n'est visiblement pas le cas. En 5.12(b) est représentée l'évolution temporelle de l'espérance conditionnelle du nombre d'occurrences de la configuration doublement phosphorylée de la protéine, sachant que exactement une des deux occurrences de la protéine est dans une configuration avec le site g phosphorylé et exactement une occurrence de la protéine est dans une configuration avec le site d phosphorylé.

autre quatruplet d'entiers naturels $\beta(\sigma) = \langle n_{\mathrm{A}} + n_{\mathrm{A}\star}, n_{\star\mathrm{A}} + n_{\star\mathrm{A}\star}, n_{\mathrm{A}} + n_{\star\mathrm{A}}, n_{\mathrm{A}\star} + n_{\star\mathrm{A}\star}\rangle$. Le vecteur $\beta(\sigma)$ est alors noté $\langle n_{\mathrm{A}\diamond}, n_{\star\mathrm{A}\diamond}, n_{\diamond\mathrm{A}}, n_{\diamond\mathrm{A}\star}\rangle$. La probabilité $P_t^\sharp(\sigma^\sharp)$ que le système soit dans un état σ tel que $\beta(\sigma) = \sigma^\sharp$ à l'instant t est définie par l'équation suivante :

$$
\begin{aligned}
\frac{dP_t^\sharp(\langle n_{\mathrm{A}\diamond}, n_{\star\mathrm{A}\diamond}, n_{\diamond\mathrm{A}}, n_{\diamond\mathrm{A}\star}\rangle)}{dt} = \\
& k^+(n_{\mathrm{A}\diamond}+1)P_t^\sharp(\langle n_{\mathrm{A}\diamond}{+}1, n_{\star\mathrm{A}\diamond}{-}1, n_{\diamond\mathrm{A}}, n_{\diamond\mathrm{A}\star}\rangle) \\
+ \quad & {}^-k(n_{\star\mathrm{A}\diamond}+1)P_t^\sharp(\langle n_{\mathrm{A}\diamond}{-}1, n_{\star\mathrm{A}\diamond}{+}1, n_{\diamond\mathrm{A}}, n_{\diamond\mathrm{A}\star}\rangle) \\
+ \quad & k^+(n_{\diamond\mathrm{A}}+1)n_{\star\mathrm{A}\diamond}P_t^\sharp(\langle n_{\mathrm{A}\diamond}, n_{\star\mathrm{A}\diamond}, n_{\diamond\mathrm{A}}{+}1, n_{\diamond\mathrm{A}\star}{-}1\rangle) \\
+ \quad & k^-(n_{\diamond\mathrm{A}\star}+1)P_t^\sharp(\langle n_{\mathrm{A}\diamond}, n_{\star\mathrm{A}\diamond}, n_{\diamond\mathrm{A}}{-}1, n_{\diamond\mathrm{A}\star}{+}1\rangle) \\
- \quad & ({}^+k n_{\mathrm{A}\diamond} + {}^-k n_{\star\mathrm{A}\diamond} + k^+ n_{\diamond\mathrm{A}} n_{\star\mathrm{A}\diamond} + k^- n_{\diamond\mathrm{A}\star})P_t^\sharp(\langle n_{\mathrm{A}\diamond}, n_{\star\mathrm{A}\diamond}, n_{\diamond\mathrm{A}}, n_{\diamond\mathrm{A}\star}\rangle) \\
- \quad & k^+\tilde{E}_t(n_{\star\mathrm{A}} \mid \langle n_{\mathrm{A}\diamond}, n_{\star\mathrm{A}\diamond}{+}1, n_{\diamond\mathrm{A}}, n_{\diamond\mathrm{A}\star}{-}1\rangle) \\
+ \quad & k^+\tilde{E}_t(n_{\star\mathrm{A}} \mid \langle n_{\mathrm{A}\diamond}, n_{\star\mathrm{A}\diamond}, n_{\diamond\mathrm{A}}, n_{\diamond\mathrm{A}\star}\rangle),
\end{aligned}
$$

dans laquelle pour toute expression $X(\sigma)$ et tout état réduit $\sigma^\sharp$, l'expression $\tilde{E}_t(X(\sigma) \mid \sigma^\sharp)$ représente le produit entre, d'une part, l'espérance conditionnelle $E_t(X(\sigma) \mid \sigma^\sharp)$ de l'expression $X(\sigma)$ sachant que l'état du système satisfait la contrainte $\beta(\sigma) = \sigma^\sharp$ et, d'autre part, la probabilité $P_t^\sharp(\sigma^\sharp)$ d'être dans un état σ tel que $\beta(\sigma) = \sigma^\sharp$.

De manière générale, l'espérance conditionnelle du nombre d'occurrences de la protéine A dans sa configuration doublement phosphorylée, sachant que l'état comporte exactement deux occurences de protéines dont exactement une dans une configuration avec le site g phosphorylé et exactement une (la même ou non) dans une

configuration avec le site d phosphorylé dépend du temps. En figure 5.12(a) est montré l'évolution du rapport entre la probabilité d'être dans l'état formé de deux occurrences de la protéine, l'une dans une configuration entièrement phosphorylée et l'autre dans une configuration entièrement non phosphorylée et la probabilité que le système soit dans un état formé de deux occurrences de la protéine, l'une dans la configuration avec le site g phosphorylé, et non le site d, et l'autre dans la configuration avec le site d phosphorylé, et non le site g. Cette courbe a été obtenue en intégrant numériquement l'équation maîtresse du système, en prenant comme distribution initiale, l'unique état formé de deux occurrences de la protéine dans sa configuration entièrement déphosphorylée et les constantes d'interaction toutes égales à 1. Non seulement, le rapport entre les deux probabilités n'est pas égal à 1, ce qui serait le cas si l'état des deux sites dans les configurations de la protéine A n'étaient pas corrélés, mais il varie avec le temps. De même, l'espérance conditionnelle du nombre d'occurrences de la protéine A dans sa configuration doublement phosphorylée sachant que le système contient exactement une occurrence de le protéine dans une configuration avec le site g phosphorylé et une occurrence de la protéine dans une configuration avec le site d phosphorylé varie avec le temps. Cette espérance conditionnelle est donnée en figure 5.12(b). On ne sait donc pas réduire l'équation maîtresse de cet exemple.

5.2.3.3 Conclusion

Cet exemple montre que, du fait que chaque occurrence de la protéine A ne peut être utilisée qu'une seule fois dans le plongement entre le membre gauche d'une règle d'interaction et l'état du système, des termes -1 peuvent apparaître dans l'équation maîtresse. Ces termes empêchent l'utilisation de la méthode de réduction qui avait été développée pour la réduction de la sémantique différentielle des modèles écrits en Kappa. Dans la sémantique différentielle, de tels termes n'apparaissent pas, car il disparaissent à la limite lorsqu'est considérée une infinité d'occurrences de chaque configuration de protéines (dans un volume infini).

5.3 Réduction de la sémantique stochastique

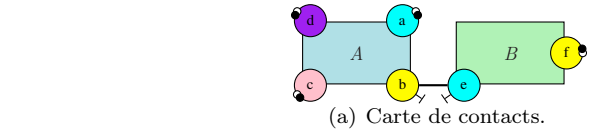
Les exemples présentés dans la section 5.2 sont riches en enseignement. Dans de rares cas, comme dans l'exemple de la section 5.2.1, il est possible de partager une protéine en deux parties. Toutefois, cela a été possible uniquement dans le cas où les configurations de ces sous-parties ne sont pas corrélées dans les occurrences de la protéine. Dans l'exemple suivant (en section 5.2.2), il a été vu qu'il est en général impossible de faire abstraction de la corrélation entre les occurrences des configurations de parties d'espèces biochimiques de part et d'autre d'une liaison entre deux sites. En effet, cette corrélation est nécessaire pour connaître la distribution des occurrences de configurations des espèces biochimiques produites quand cette liaison est détruite. De la même manière, il n'est pas possible de faire abstraction de la corrélation entre les occurrences des configurations de deux parties d'espèces biochimiques qui se chevauchent, car toute règle transformant la partie commune transformeraient deux occurrences de configurations de parties à la fois, ce qui nécessite d'en connaître la corrélation. Enfin, plus surprenant, l'exemple de la section 5.2.3 a montré qu'il pouvait être nécessaire de connaître la corrélation entre l'état de deux sites dans les occurrences des configurations d'une protéine, même si ces deux sites n'apparaissent dans le membre gauche de règles que dans des occurrences de protéines différentes.

Dans cette section est proposée une analyse statique pour réduire la sémantique stochastique des modèles de réécriture de graphes à sites, en tenant compte de ses observations.

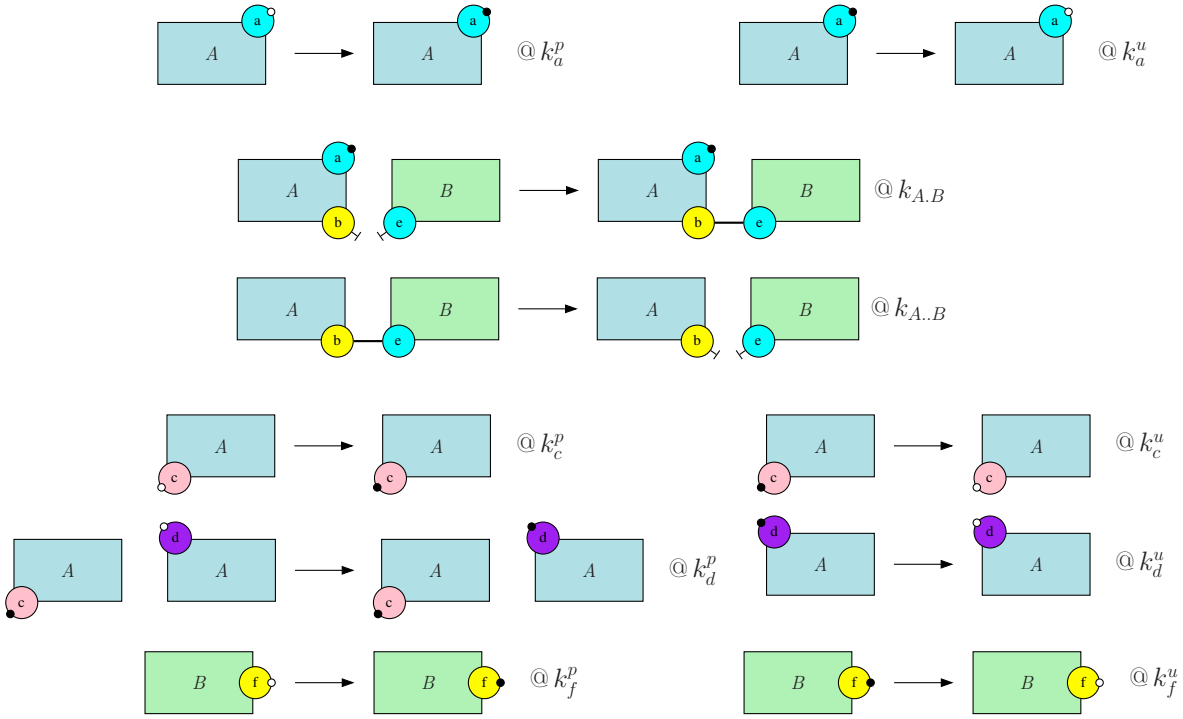
5.3.1 Analyse statique

Cette approche consiste à découper les occurrences des configurations de protéines en des parties disjointes en regardant quelles paires de sites apparaissent dans une même règle (pas nécessairement dans la même occurrence d'une protéine). Les liens entre sites, eux, sont gardés tels quels. Ainsi dans la carte de contacts du modèle, un arc bidirectionnel sera dessiné entre deux sites d'une même protéine dès lors qu'il existe une règle comportant ces deux sites dans des occurrences de la protéine en question.

Exemple 5.3.1 *Cette analyse est illustrée sur un exemple jouet. Celui-ci comprend deux protéines A et B. Chaque occurrence de la protéine A comporte* 4 *sites, a, b, c et d, alors que chacune de la protéine B comporte* 2 *sites, e et f. Comme indiqué dans la carte de contacts, en figure 5.13(a), Les sites b et e sont des sites de liaisons : les occurrences de ces sites peuvent être libres et une occurrence du site b peut être liée à une*



(a) Carte de contacts.



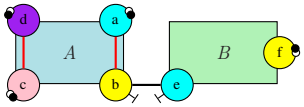
(b) Règles d'interaction.

Figure 5.13: Modèle jouet pour étudier la reduction de la sémantique stochastique des modèles Kappa. En 5.13(a), la carte de contacts introduit deux protéines, A et B. La protéine A dispose de trois sites de phosphorylation, a, c et d, et la protéine B d'un site de phosphorylation, f. La protéine A a également un site de liaison, b, qui peut se lier à un site de liaison, e, de la protéine B. En 5.13(b) sont données les règles d'interaction de ce modèle. Les sites a et c de la protéine A et f de la protéine B peuvent se phosphoryler et se déphosphoryler sans conditions. La phosphorylation du site a de la protéine A contrôle sa capacité à se lier à la protéine B. Ainsi seules les occurrences de la protéine A dans une configuration dans laquelle le site a est phosphorylé (et le site b libre) peuvent se lier aux occurrences de la protéine B (dans une configuration avec le site e libre). La dissociation correspondante peut se faire sans conditions. Enfin la phosphorylation du site d d'une occurrence de la protéine A requiert que d'autres occurrences de la protéine A soient dans une configuration avec le site c phosphorylé. La déphosphorylation correspondante se fait sans conditions. Chaque interaction a sa propre constante d'interaction.

occurrence du site e. Les autres sont des sites de phosphorylation : leurs occurrences peuvent phosphorylées (ce qui est représenté par une pastille noire), ou non (ce qui est représenté par une pastille blanche).

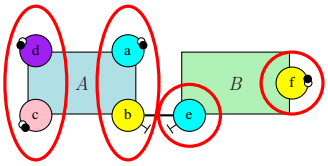
Les règles d'interactions sont données en figure 5.13(b). Les deux premières décrivent la phosphorylation et la déphosphorylation du site a des occurrences de la protéine A, Ces deux interactions sont purement locales. Elles ont pour constantes d'interaction respectivement k_a^p *et* k_a^u*. Suivent les règles d'association et de dissociation entre le site b des occurrences de la protéine A et le site e des occurrences de la protéine B. L'association entre les deux protéine est controlée par le site a de la protéine A dont seules les occurrences dans une configuration dans laquelle le site a est phosphorylée peuvent se lier. La constante d'association est notée* $k_{A.B}$ *et celle de dissociation* $k_{A..B}$*. Ensuite sont données les règles de phosphorylation et déphosphorylation du site c des occurrences de la protéine A, Ces interactions peuvent s'effectuer sans conditions. Elles ont respectivement pour constantes d'interaction* k_c^p *et* k_c^u*. L'état du site c des occurrences de la protéine A exerce un contrôle à distance sur la phosphorylation du site d des autres occurrences de cette même protéine. Ainsi le taux de phosphorylation du site d dans une occurrence de la protéine A est proportionnel au nombre des autres occurrences de la protéine A dans une configuration dans laquelle le site c est phosphorylé. La constante de cette interaction est notée* k_d^p*. La déphosphorylation correspondante se fait sans conditions avec une constante d'ineraction* k_d^u*. Enfin sont dessinées les règles de phosphorylation et déphosphorylation du site f des occurrences de la protéine B. Ces interactions peuvent s'effectuer sont inconditionnelles. Elles ont respectivement pour constantes d'interaction* k_f^p *et* k_f^u*.*

Deux règles induisent des corrélations entre les états des sites des protéines. D'une part, la règle de liaison induit une corrélation entre l'état du site a et l'état du site b dans les occurrences des différentes configurations de la protéine A. Par ailleurs, la règle de phosphorylation du site d induit une corrélation entre l'état du site c et l'état du site d. Il en résulte l'annotation suivante de la carte de contacts :



Les arcs sont utilisés pour réaliser, pour chaque protéine, une partition de l'ensemble de ses sites d'interaction. Cette partition s'obtient en calculant les composantes connexes (qui regroupent toute paire de sites reliés par un chemin d'arcs) de ces arcs.

Exemple 5.3.2 *Dans l'exemple jouet, la partition des sites de la protéine A et celle de ceux de la protéine B sont dessinées dans la carte de contacts suivante :*



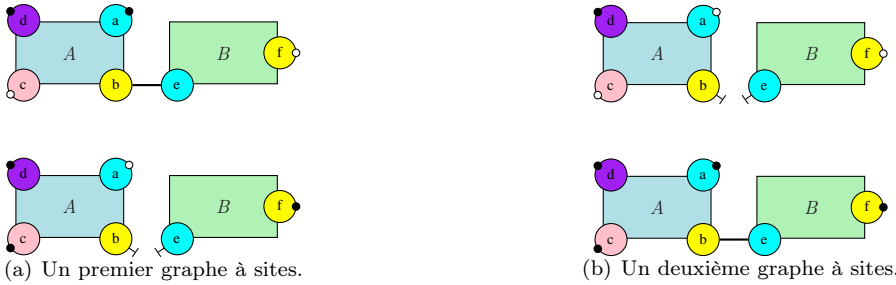
5.3.2 Réduction de modèles

Il est possible d'utiliser l'annotation de la carte de contacts d'un modèle pour en réduire la sémantique stochastique. Cette réduction peut être décrite directement en Kappa, en transformant le modèle initial en un autre modèle. Elle consiste essentiellement à découper les occurrences de configurations de protéines dans les graphes à sites qui décrivent l'état du système, selon les partitions spécifiées dans la carte de contacts annotée et à spécialiser les règles pour qu'elles agissent sur les bonnes parties des protéines.

5.3.2.1 Abstraction d'un graphe à sites

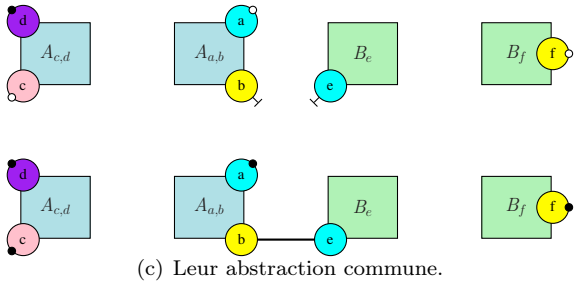
Le but de cette section est de définir l'abstraction d'un graphe à sites étant donné une carte de contacts annotée.

Les configurations associées aux occurrences des parties d'une protéine obtenues en découpant les occurrences des configurations de cette protéine comme spécifié dans la carte de contacts se comportent de manière indépendante. L'abstraction d'un graphe à sites consiste donc à découper chaque occurrence de configurations de protéines dans ce graphe à sites selon ces partitions. Pour distinguer les différentes parties des protéines, les noms de protéines sont associés à l'ensemble ordonnée des sites qui constituent la classe d'équivalence correspondante dans la carte de contacts annotée. En particulier, deux graphes à sites obtenus en échangeant dans l'un l'état des sites d'une classe d'équivalence entre deux occurrences de configurations de protéine auront la même abstraction.



(a) Un premier graphe à sites.

(b) Un deuxième graphe à sites.

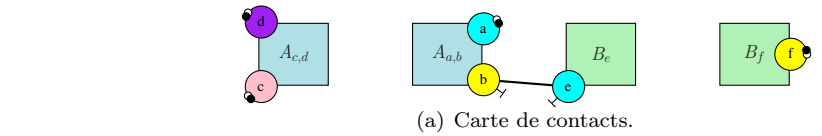


(c) Leur abstraction commune.

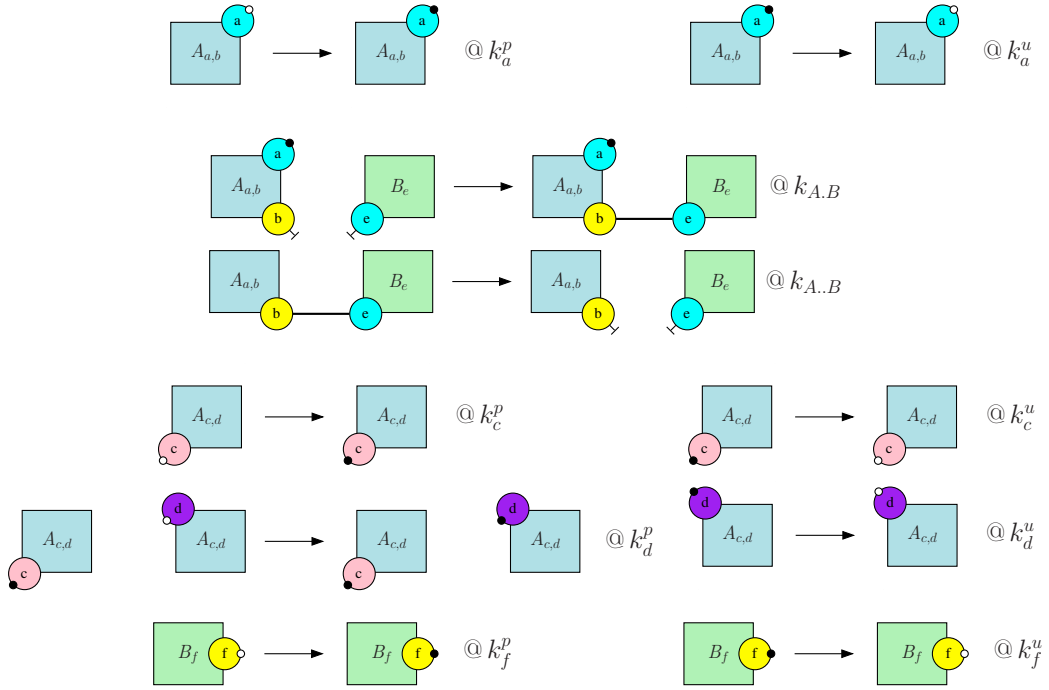
Figure 5.14: Deux graphes à sites qui ont la même abstraction en découpant leurs occurrences de configurations de protéines suivant les partitions figurant dans la carte de contacts annotée. Les deux graphes à sites sont donnés en 5.14(a) et 5.14(b). L'abstraction de chaque graphe à sites est obtenue en découpant chaque occurrence de configurations de la protéine A avec les sites a et b d'un côté et les sites c et d de l'autre. Les occurrences de configurations de la protéine B sont, elles, découpées en deux en séparant l'état du site e de celui du site f. Dans les deux cas, le graphe à sites obtenu est celui qui est dessiné en 5.14(c). En effet, les deux graphes de départ peuvent être obtenus l'un à partir de l'autre en échangeant simultanément l'état des sites a et b entre les deux occurrences de configurations de la protéine A et l'état du site e entre les deux occurrences de configurations de la protéine B.

Exemple 5.3.3 *L'abstraction de graphes à sites est illustrée dans l'exemple jouet. En figure 5.14, sont donnés deux exemples de graphes à sites qui ont la même abstraction. En figure 5.14(a) est dessiné un graphe à sites comportant deux occurrences de la protéine A et deux occurrences de la protéine B. La première occurrence de la protéine A est dans une configuration dans laquelle les sites a et d sont phosphorylées et le site c ne l'est pas. Son site b est lié à une occurrence de configurations de la protéine B dans laquelle le site f n'est pas phosphorylé. La deuxième occurrence de la protéine A est dans une configuration dans laquelle les sites c et d sont phosphorylés, mais pas le site a. Son site b est libre. La deuxième occurrence de la protéine B est dans une configuration dans laquelle le site e est libre et le site f phosphorylé. Le deuxième graphe à sites, qui est dessiné en figure 5.14(b), s'obtient à partir du premier en échangeant l'état des sites c et d dans les deux occurrences de configurations de la protéine A, ainsi que l'état du site e dans les deux occurrences de configurations de la protéine B.*

Comme les sites c et d forment une composante connexe dans la carte de contacts annotée, ainsi que le site e, ces échanges de sites ne changent pas l'abstraction du graphe à sites. Cette abstraction est représentée en figure 5.14(c). Elle contient deux occurrences de la partie gauche de la protéine A, notée $A_{c,d}$, l'une dans sa configuration avec le site c non phosphorylé et le site d phosphorylé et l'autre dans sa configuration avec ses deux sites phosphorylés. Elle contient également deux occurrences de la partie droite de la protéine A, notée $A_{a,b}$, l'une dans sa configuration avec son site a non phosphorylée et son site b libre et l'autre dans sa configuration avec le site a phosphorylé et le site b lié à une occurrence de configurations de la partie gauche de la protéine B, notée B_e. Il y a une autre occurrence de la partie gauche de la protéine B dans une configuration avec le site e libre. Elle contient enfin deux occurrences de la partie droite de la protéine B, notée B_f, l'une dans la configuration dans laquelle le site f est phosphorylée, et l'autre non.



(a) Carte de contacts.



(b) Règles d'interaction.

Figure 5.15: Modèle réduit pour le modèle jouet de figure 5.13. Le modèle comporte quatre parties de protéine, chacune étant représentée par le nom de la protéine initiale avec en index la liste des sites qui constituent la partie en question. La carte de contacts, donnée en 5.15(a), comporte donc quatre morceaux de protéines, $A_{a,b}$, $A_{c,d}$, B_e et B_f. En 5.15(b), sont données les règles du modèle réduit. En particulier, le nom des protéines qui apparaissent dans les règles est remplacé par celui correspondant dans le modèle réduit en ajoutant en indice les sites qui constituent la classe d'équivalence sur laquelle agit la règle correspondante. Les règles gardent les mêmes constantes d'interactions.

5.3.2.2 Abstraction d'un ensemble de règles

Par construction, les sites qui apparaissent dans les occurrences d'une protéine dans une règle donnée, appartiennent tous à la même classe d'équivalence, et donc à la même partie de cette protéine. Ainsi, l'abstraction d'une règle s'obtient en remplaçant le nom des protéines qui apparaissent dans la règle par le nom de la portion correspondante. Il suffit ainsi d'annoter le nom de la protéine par l'unique classe d'équivalence qui contient les sites utilisés dans la règle. Le cas où l'occurrence d'une protéine ne mentionne aucun site est particulier, car toutes les classes d'équivalence peuvent convenir. Il suffit alors d'en choisir une de manière arbitraire.

Exemple 5.3.4 *La réduction de l'ensemble de règles donné en figure 5.13 est donnée en figure 5.15. Le nom de chaque protéine dans les occurrences de configurations de protéines est remplacé par le nom correspondant dans le modèle réduit en ajoutant en indice la liste dans l'ordre alphabétique des sites qui constituent la classe d'équivalence correspondante.*

Pour les règles locales, chaque occurrence de protéines ne documente qu'un site. Ce site est nécessairement dans une unique classe d'équivalence. Pour la règle d'association, l'état du site a exerce un contrôle sur l'état du site b. Ceci assure qu'ils sont dans la même classe d'équivalence. Il existe donc bien une classe d'équivalence qui les contient tous les deux. Enfin, pour la règle de phosphorylation du site d, l'état du site c exerce un contrôle à distance. De ce fait, les sites c et d apparaissent dans la même classe d'équivalence. C'est celle-ci qui est

utilisée pour remplacer les deux occurrences de la protéine A. Si des noms de protéines différents avaient été utilisés dans cette règle, celle-ci aurait engendré plus de pas de calculs potentiels dans le modèle réduit que la règle initiale.

5.3.2.3 Impact sur la sémantique stochastique

Les sémantiques stochastiques du modèle initial et du modèle réduit sont fortement liées.

En effet, les pas de calculs issus d'un état du modèle initial et les pas de calculs issus de son abstraction dans la modèle réduit sont en bijection. Il suffit d'appliquer l'abstraction de la règle correspondante pour obtenir un pas de calcul entre l'abstraction de l'état avant le pas de calcul et l'abstraction de l'état après celui-ci. Par ailleurs, la propensité du pas de calcul dans la modèle initial et celle du pas de calcul correspondant dans le modèle réduit est la même (les deux pas de calculs viennent de la même règle).

Ceci se résume dans le diagramme commutatif suivant :

$$\begin{array}{ccc} q & \xrightarrow{r} & q' \\ {\scriptstyle\beta}\downarrow & & \downarrow{\scriptstyle\beta} \\ \beta(q) & \xrightarrow{\beta(r)} & \beta(q') \end{array}$$

dans lequel $\beta(q)$ et $\beta(q')$ représentent respectivement les abstractions des états q et q', et $\beta(r)$ l'abstraction de la règle r. Les flèches verticales représentent les abstractions des états alors que les flèches horizontales décrivent les pas de calcul engendrés par les règles (que ce soit dans le modèle initial ou dans le modèle reduit).

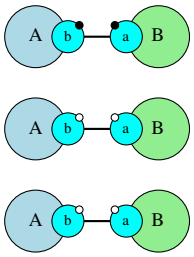
La relation qui identifie tous les états ayant la même abstraction est appelée une bisimulation avant-arrière. En particulier, l'abstraction a les propriétés d'une bisimulation en avant. Ainsi, les ensembles de traces du modèle réduit ont même probabilité que la somme des probabilités des ensembles de traces du modèle initial dont ils sont l'abstraction [76, 75, 116] (l'abstraction d'une trace est obtenue en appliquant la fonction d'abstraction à chaque état et à chaque pas de calculs de la trace tout en préservant les intervalles de temps). Lors d'une simulation, il est possible de remplacer l'état du système par un état ayant la même abstraction, sans changer le comportement du système vis à vis des ensembles d'états qui ont la même abstraction. En ce qui concerne l'équation maîtresse, la probabilité d'être dans un état abstrait à un instant donné, est la somme des probabilités d'être dans un état dont c'est l'abstraction, à cet instant [22].

La relation est également une bisimulation arrière. Cela signifie qu'elle préserve des rapports de proportionnalité. Ainsi, si la distribution des états initiaux est telle que pour chaque paire d'états q et q' ayant la même abstraction par la fonction β, les probabilités d'être dans l'état q et q' sont inversement proportionnelles à leurs nombres d'automorphismes respectifs, alors cette propriété est préservée au cours de l'exécution de la sémantique stochastique. En particulier, dans la sémantique de traces [76, 75, 116], cela signifie que si cette propriété est vérifiée dans la distribution des états initiaux, alors c'est le cas après avoir appliquer des pas de calculs engendré par la même suite de règles et dans les mêmes intervalles de temps. En ce qui concerne l'équation maîtresse, cette propriété est un invariant. Si elle est satisfaite pour $t = 0$, elle reste vraie à tout instant de l'exécution [22].

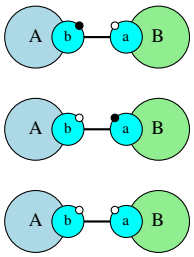
Le rôle du nombre d'automorphismes dans les rapports de proportionnalité peut surprendre. Il faut pour comprendre son origine raisonner sur des traces dans lesquelles chaque occurrence de protéines porte un identifiant unique. Cette notion de trace donne une observation plus fine, puisqu'un état classique peut être obtenu en oubliant les identifiants dans un état annoté ou encore vu comme la classe d'équivalence d'un état annoté quitte à échanger les identifiants des occurrences des protéines de la même sorte (qui peuvent par contre être dans des configurations différentes). La bisimulation en arrière assure que si pour chaque paire d'états obtenus l'un à partir de l'autre en échangeant les identifiant des occurrences des protéines de la même sorte est équiprobable dans la distribution initiale des états, alors cette propriété reste vraie au cours de l'exécution de la sémantique stochastique. Le nombre d'états avec identifiant qui donne le même état classique, n'est pas toujours le même. En conséquence tous les états doivent avoir un poids proportionnel au nombre d'états avec identifiants qui lui correspondent dans la sémantique classique. Or ce nombre est inversement proportionnel au nombre d'automorphismes, d'où le rôle de nombre des automorphismes dans les rapports de proportionnalité.

Exemple 5.3.5 *L'exemple suivant illustre la relation entre le nombre d'automorphismes dans un graphe à sites et le nombre de manières d'obtenir un état isomorphe en permutant les identifiants de ses occurrences de protéines.*

Cet exemple comporte deux protéines A et B. Chaque protéine comporte un site de phosphorylation et un site de liaison pouvant se lier au site de liaison de l'autre protéine. Le but est de comprendre l'action des permutations d'identifiants dans un état formé de trois occurrences du dimère formé par les protéines A et B, une dans sa configuration doublement phosphorylée et deux dans sa configuration entièrement non phosphorylée. Cet état est représenté ci-dessous.



Il y aura donc $2 \cdot 3!$ *permutations d'identifiants possible.* $3!$ *pour les occurrences de la protéine A et* $3!$ *pour les occurrences de la protéine B. En appliquant ces transformations, certaines vont laisser l'état dans une situation isomorphe avec une occurence du dimère dans la configuration doublement phosphorylé et deux occurrences du dimère dans la configuration sans phosphorylation. Cet état admet* 2 *automorphismes : l'automorphisme trivial et celui qui consiste à échanger les deux occurrences du dimère dans une configuration sans phosphorylation. D'autres transformations vont donner l'état suivant :*



dans lequel une occurrence du dimère est dans la configuration dans laquelle l'occurrence de la protéine A est phosphorylée, et non celle de la protéine B, une autre dans la configuration dans laquelle l'occurrence de la protéine B est phosphorylée, et non celle de la protéine A et une autre dans la configuration sans phosphorylation. Cet état ne comporte que l'automorphisme trivial.

Exactement $2 \cdot 3!$ *permutations d'identifiants laissent le premier état dans un état isomorphe. Il est possible de choisir la permutation des identifiants des occurrences de la protéine A de manière arbitraire, ce qui donne le facteur* $3!$*. Ensuite, il n'y a qu'une possibilité pour l'occurrence de la protéine B dans une configuration phosphorylée puisque qu'elle doit être liée à l'occurrence de la protéine A dans une configuration phosphorylée. Ensuite, il est possible d'échanger ou non les deux occurrences de la protéine B dans une configuration non phosphorylée, ce qui donne le facteur* 2*.*

Exactement $2 \cdot 2 \cdot 3!$ *permutations changent le deuxième état dans un état isomorphe. En effet, n'importe quelle permutation peut être utilisée pour les occurrences de la protéine A ce qui donne la facteur* $3!$*. Il y a deux choix pour l'occurrence de la protéine B dans sa configuration phosphorylée, puisqu'elle doit être liée à l'une ou l'autre des occurrences de la protéine A dans une configuration non phosphorylée, ce qui donne un facteur* 2*. Il reste alors à choisir librement les identifiants des deux autres occurrences de la protéine B ce qui donne un deuxième facteur* 2*.*

Aucune permutation n'a été oubliée, puisque $2 \cdot 3! + 2 \cdot 2 \cdot 3! = 3 \cdot 2 \cdot 3! = 2 \cdot 3!$*. Par ailleurs, l'équation suivante :*

$$(2 \cdot 3!)2 = (2 \cdot 2 \cdot 3!)$$

est bien vérifiée, ce qui illustre bien le fait que le nombre d'automorphismes est inversement proportionnel au nombre de permutations d'identifiants qui laissent l'état isomorphe.

5.4 Pour aller plus loin

Dans ce chapitre, des exemples bien choisis ont permis d'illustrer la difficulté de réduire, de manière exacte, la sémantique stochastique des modèles de réécriture de graphes à sites. En fait, l'effet de l'application des

Modèle	egfr (simplifié)	Interactions entre les voies EGF et celles de l'insuline	egfr, erk, mapk, ras
Nombre de configurations d'espèces biochimiques	356	2899	$\sim 2.10^{19}$
Nombre de fragments en différentiel	38	208	$\sim 2.10^{5}$
Nombre de fragments en stochastique	356	618	$\sim 2.10^{19}$

Figure 5.16: Puissance de réduction pour la sémantique différentielle (voir en chapitre 4) et pour la sémantique stochastique (ce chapitre). L'approche a été testée sur trois modèles : un modèle simplifié des événements précoces dans l'acquisition du facteur de l'épiderme, un modèle d'interférences entre les voies d'acquisition du facteur de l'épiderme et celles de l'insuline (issu de [39, table 7]), et le modèle plus complet de la voies de signalisation du facteur de croissance de l'épiderme qui inclue plusieurs étapes après le recrutement des occurrences de la protéine *Sos* [51, 10, 124, 20]. La réduction de la sémantique différentielle du premier et du troisième modèle avait déjà été testé dans le chapitre 4.

règles dans la sémantique stochastique dépend beaucoup plus de la corrélation entre l'états des sites dans les occurrences des configurations des protéines que pour la sémantique différentielle. En particulier, quand des portions de protéines se chevauchent ou quand deux protéines sont liées, toute modification de la configuration de la partie commune ou toute dissociation du lien modifie conjointement plusieurs portions de protéines. Il est alors nécessaire de connaître le corrélation entre les occurrences des configurations de ces portions de protéines (voir en section 5.2.2). Par ailleurs, quand deux occurrences d'une même protéine apparaissent dans une même règle, il faut pouvoir vérifier que la règle s'applique à deux occurrences distinctes de la protéine, ce qui nécessite également de connaître la corrélation entre les propriétés testées par chacune des occurrences de la règle dans les occurrences des configurations de la protéine (voir en section 5.2.3).

La description de l'approche dans [117] permet de l'appliquer à tout le language Kappa. En particulier, elle s'applique aussi aux règles qui comportent des créations et de dégradations d'occurrences de protéines. Toutefois, le résultat de bisimulation est moins fort dans ce contexte. En effet, une règle avec des créations d'occurrences de protéines peut créer des corrélations entre l'état des sites d'interactions dans les occurrences des configurations des protéines. Ceci laisse deux possibilités : soit ne pas découper l'ensemble des sites des protéines dont les occurrences peuvent être créées par les règles, soit perdre les invariants statistiques et n'utiliser que des bisimulations en avant. Dans le cas de dégradations d'occurrences de protéines, le comportement des règles de dégradation dépend de la corrélation entre l'états des sites dans les occurrences des configurations des protéines dégradées. De ce fait, ceci laisse deux possibilité : soit ne pas découper l'ensemble des sites des protéines dont les occurrences peuvent être dégradées ou se contenter de bisimulations en arrière qui ne permettent de réduire la sémantique stochastique que pour les distributions d'états initiales qui vérifient les bons rapports de proportionnalités.

Dans ce rare cas, deux parties d'une protéine agissent de manière entièrement indépendante (voir en section 5.2.1). Dans ce cas, il est possible de découper les occurrences de configurations des protéines correspondantes en portion sans parties communes. Il est alors possible de générer un nouveau modèle Kappa opérant sur les portions de protéine ainsi obtenue. En figure 5.16 sont données les nombres de configurations d'espèces biochimiques, le nombre de variables dans le système différentiel réduit avec la méthode du chapitre 4 et le nombre de configurations d'espèces biochimiques dans le modèle Kappa réduit avec la méthode du présent chapitre. En pratique, aucune simplification n'est apportée par la présente méthode, sauf dans le deuxième modèle. Dans ce cas, la réduction provient de deux parties de la protéine *Sos* qui agissent de manière complètement indépendante, l'une étant activée par le facteur de croissance *EGF*, l'autre par l'insuline.

Cela met en lumière l'interêt de la sémantique stochastique : contrairement à la sémantique différentielle, elle ne se contente pas de décrire un comportement à la limite, quand les quantités de composants tendent vers l'infini. Elle permet aussi de raisonner sur la variabilité d'un système et de sa robustesse vis à vis des différents événements stochastiques. En contre-partie, il est très difficile de simplifier cette sémantique de manière exacte.

Encore plus que pour la réduction de la sémantique différentielle, le critère d'exactitude dans le cadre stochastique est bien trop fort. Il existe différentes pistes pour palier à cette limite. D'une part, les notions de bisimulations ont été assouplies. Un point faible des bisimulations est qu'elles ne guarantissent aucune propriété sur les paires d'états qui ne sont pas en bisimulation. C'est tout ou rien. Les métriques de bisimlations [63, 80] ont été introduites pour quantifier graduellement les comportements de deux programmes ou de deux états (deux programmes ou deux états à distance 0 seront alors bisimilaires). Il reste toutefois difficile d'en extrapoler des propriétés quantiatives sur la sémantique de traces ou la solution de l'équation maîtresse. D'autre part, il

est possible, au lieu de laisser la méthode imposer comment les occurrences de configurations de protéines sont découpées, de permettre au concepteur de modèles de définir lui-même les partitionnements. Ceci permettrait également de passer sous silence des corrélations qui n'influeraient que de manière marginale sur la dynamique du système. Quelles en seraient les conséquences ? Dans le cas général, la connaissance de l'état du système ne sera pas suffisante pour caractériser les distributions pour le délai entre deux pas de calculs et l'effet de ces pas de calculs sur l'évolutions du nombre d'occurrences des motifs d'intérêts. Il en résultera des pertes d'information à la fois en terme de probabilités, du nombre d'occurrences des différents motifs et aussi du temps écoulé. Sans ce contenter de résultats approchés numériquement et suivant les principes de l'interprétation abstraite, toute perte d'information sera encadrée par dessous et par dessus. Ceci permettra d'interpréter les résultats en terme du système initial au risque qu'une trop grande perte d'information empêche de répondre aux questions intéressantes - rien de faux ne sera prouver, mais certaines questions resteront sans réponse du fait de l'imprécision de l'abstraction. De ce fait, seuls des systèmes robustes auront des dynamiques assez stables pour éviter que ces pertes d'information ne s'accumulent trop le long de l'exécution réduite, mais approchée du système.

Chapitre 6

Symétrie dans les graphes à sites

Les symétries apparaissent sous des formes diverses dans les modèles décrits par des règles de réécriture. Certaines sont indissociables de la sémantique. C'est le cas des permutations entre occurrences de protéines qui laissent les graphes à sites inchangés. D'autres peuvent décrire des équivalences entre sites d'interaction, des équivalences entre sortes de protéines ou encore des équivalences spatiales qui permettent de voir la structure de certaines espèces biochimiques modulo des isométries.

Dans [73], nous avons introduit un cadre de travail algébrique unifiant pour décrire, inférer et utiliser des groupes de transformations de graphes à sites dans le langage Kappa. L'idée principale est de considérer des groupes de transformations qui agissent sur les graphes à sites avec une opération supplémentaire permettant de restreindre une transformation sur les sous-graphes d'un graphe. Il est alors possible de définir quand un ensemble de règle est symétrique par rapport à un groupe de transformations et d'en déduire des propriétés au niveau du comportement global du modèle, que ce soit pour sa sémantique stochastique ou différentielle. Pour simplifier la présentation, nous nous concentrons ici aux symétries qui correspondent à des échanges de sites, qui sont décrites dans [27].

Travaux voisins. Plusieurs formalismes permettent de faire apparaître chaque site plusieurs fois dans l'interface des agents. C'est le cas du langage BNGL [9]. Dans le langage ReactC [100], des sites indistinguables peuvent être encodés à l'aide d'hyperliens. Ceci procure un moyen syntaxique de décrire des sites symétriques. En contre-partie, le calcul des ensembles des occurrences des motifs de réécriture devient prohibitif (les graphes ne sont plus rigides), ce qui pose de grands problèmes de complexité pour échantillonner les trajectoires de la sémantique stochastique.

En Kappa, la rigidité des graphes est un principe fondamental, et donc, l'utilisation de plusieurs occurrences d'un même site dans l'interface d'un agent est interdite. Les équivalences entre sites d'interaction peuvent toutefois être décrites à un plus haut-niveau de spécification [91] puis traduites automatiquement dans le noyau pure du langage Kappa. Cependant, il faut être conscient que cette traduction peut générer un très grand nombre de règles, là où une seule règle aurait été suffisante en utilisant des sites à occurrences multiples.

Dans ce chapitre, nous proposons une approche pour détecter automatiquement des équivalences entre sites d'interaction. Ces équivalences n'ont pas besoin d'être spécifiées à la main. Puis, nous utilisons ces équivalences pour réduire l'espace d'états ou le nombre de configurations des espèces biochimiques dans les modèles considérés. Ceci permet de réduire la dimension des systèmes différentiels sous-jacents et facilite le calcul de propriétés sur la distribution des traces de la sémantique stochastique. Ce n'est en revanche pas crucial pour l'échantillonnage des traces de la sémantique stochastique, puisque la simulation travaille directement au niveau des graphes à sites [54].

Enfin, notre cadre de travail n'est pas limité à la prise en compte des équivalences entre sites d'interaction. Il peut être utilisé pour considérer des chaînes d'agents indépendamment de leurs orientations ou des sous-groupes d'échanges de sites comme c'est souvent le cas en chemoinformatique.

Remerciements.

Ce chapitre reprends donc les principaux résultats qui avaient été établis avec Ferdinanda Camporesi [27] à propos de la détection et de la prise en compte d'équivalences entre sites d'interaction dans un noyau sans effets

de bord du langage Kappa, pour réduire la combinatoire de ses modèles. L'extension à des groupes de symétries arbitraires et au langage complet qui est introduite dans [73] n'est évoquée qu'en conclusion de ce chapitre.

6.1 Le cas d'études

Nous illustrons la notion de symétries dans Kappa à travers un exemple jouet.

6.1.1 Modèle

Nous considérons un example dans lequel certains sites d'une protéine ont exactement les mêmes capacités d'interaction.

6.1.1.1 Les constituants du modèle et ses règles d'interaction

On considère une unique sorte de protéine, A. On suppose que chaque occurrence de cette protéine dispose exactement de deux sites de liaison qui seront notés x et y. Les sites de deux occurrences différentes de la protéine A peuvent se lier arbitrairement de manière réversible. Il y a donc trois types de liaison, x-x, x-y et y-y, selon les sites qui sont impliqués dans la liaison. De plus, il est fait l'hypothèse que dans chaque occurrence de cette protéine, au plus un site peut être lié à la fois. Sous ces hypothèses, il existe exactement quatre configurations d'espèces biochimiques[1]. Celles-ci sont toutes les quatre dessinées en figure 6.1. L'occurrence d'une configuration d'espèces biochimiques est ainsi formée soit d'une occurrence de la protéine A avec les deux sites libres (voir en figure 6.1(a)), soit de deux occurrences de la protéine A liées par leurs deux sites x (voir en figure 6.1(b)), par leurs deux sites y (voir en figure 6.1(c)) ou par le site x de l'une et le site y de l'autre (voir en figure 6.1(d)).

Les règles du modèle sont décrites en figure 6.2 et en figure 6.3. La figure 6.2 contient trois règles de liaisons. La règle dessinée en figure 6.2(a) stipule que deux occurrences de la protéine A dont tous les sites sont libres peuvent lier leurs sites x respectifs avec la constante de réaction k_{xx}. La règle donnée en figure 6.2(b) précise que deux occurrences de la protéine A dont tous les sites sont libres peuvent lier leurs sites y respectifs avec la constante de réaction k_{yy}. La règle dessinée en figure 6.2(c) spécifie que deux occurrences de la protéine A dont tous les sites sont libres, peuvent se lier respectivement par le site x de l'une et le site y de l'autres avec la constante de réaction k_{xy}. Ces différents types de liens peuvent être brisés. Les règles de dissociation sont décrites en figure 6.3. Selon la règle donnée en figure 6.3(a), un lien entre deux sites x peut être brisé avec une constante de réaction k^d_{xx}. Selon la règle dessinée en figure 6.3(b), un lien entre deux sites y peut être brisé avec une constante de réaction k^d_{yy}. Enfin, le règle décrite en figure 6.3(c), un lien entre deux sites x et y peut se casser avec une constante de réaction k^d_{xy}.

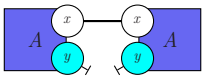
6.1.1.2 Le comportement du modèle

D'un point de vue qualitatif, les sites x et y des occurrences de la protéine A ont exactement les mêmes capacités d'interaction. On peut étudier de manière empirique ce que cela implique au niveau de la sémantique du modèle, et ce pour les deux types de sémantiques quantitatives, à savoir la sémantique stochastique et la sémantique différentielle.

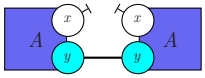
6.1.1.2.1 Équation maîtresse. Pour regarder le comportement de la sémantique stochastique du modèle, nous proposons de nous concentrer sur la solution de son équation maîtresse [111]. Celle-ci décrit pour chaque état potentiel du système sous-jacent, l'évolution temporelle de la probabilité que le système soit dans cet état. Comme il y a quatre configurations d'espèces biochimiques, l'état du système sera représenté par un vecteur de quatre entiers naturels. Ainsi la variable $q_{i,j,k,l}$ représente l'état dans lequel il y a exactement i occurrence(s) de la protéine A sans sites liés (voir en figure 6.1(a)), j paires d'occurrences de la protéine A liées par leurs deux sites x respectifs (voir en figure 6.1(b)), k paires d'occurrences de la protéine A liées par leurs deux sites y respectifs (voir en figure 6.1(c)) et l paires d'occurrences de la protéine A liées l'une par son site x et l'autre par son site y (voir en figure 6.1(d)). L'équation maîtresse nécessite une variable par état potentiel. Pour faire tenir les équations sur une seule page, il est nécessaire de ne considérer qu'un nombre très limité d'occurrences de la protéine A. De ce fait, le système débutera avec probabilité 1 dans un état avec uniquement six occurrences de la

[1] voir la section 2.2 pour la notion de configuration d'une espèce biochimique.

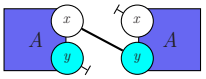
(a) Une occurrence de la protéine A avec ses deux sites de liaison libres.



(b) Deux occurrences de la protéine A liées par leurs deux sites x.

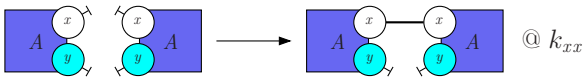


(c) Deux occurrences de la protéine A liées par leurs deux sites y.

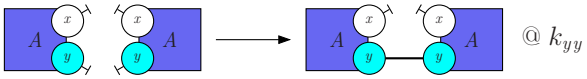


(d) Deux occurrences de la protéine A liées respectivement par le site x de l'une et le site y de l'autre.

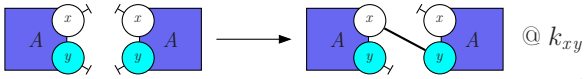
Figure 6.1: Les quatre configurations d'espèces biochimiques du cas d'étude sur les symétries.



(a) Règle de liaison entre les sites x de deux occurrences de la protéine A dont le site y est libre.

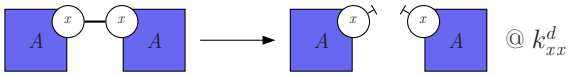


(b) Règle de liaison entre les sites y de deux occurrences de la protéine A dont le site x est libre.

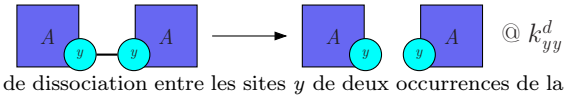


(c) Règle de liaison entre le site x d'une occurrence de la protéine A et le site y d'une autre occurrence de la protéine A, sous la condition que le site y de la première occurrence de protéine et le site x de la seconde soient tous les deux libres.

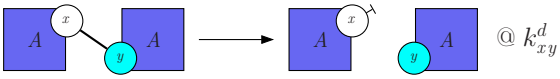
Figure 6.2: Les trois règles de liaison du cas d'étude sur les symétries.



(a) Règle de dissociation entre les sites x de deux occurrences de la protéine A dont le site y est libre.



(b) Règle de dissociation entre les sites y de deux occurrences de la protéine A dont le site x est libre.



(c) Règle de dissociation entre le site x d'une occurrence de la protéine A et le site y d'une autre occurrence de la protéine A.

Figure 6.3: Les trois règles de dissociation du cas d'étude sur les symétries.

$$
\begin{cases}
\frac{\mathrm{d}P_t(q_{6,0,0,0})}{dt} = 2k^d_{xx}P_t(q_{4,1,0,0}) + 2k^d_{yy}P_t(q_{4,0,1,0}) + k^d_{xy}\cdot P_t(q_{4,0,0,1}) - 30(k_{xx}+k_{yy}+k_{xy})P_t(q_{6,0,0,0}) \\
\frac{\mathrm{d}P_t(q_{4,1,0,0})}{dt} = 30k_{xx}P_t(q_{6,0,0,0}) + 4k^d_{xx}P_t(q_{2,2,0,0}) + 2k^d_{yy}P_t(q_{2,1,1,0}) + k^d_{xy}\cdot P_t(q_{2,1,0,1}) - \left(12(k_{xx}+k_{yy}+k_{xy}) + 2k^d_{xx}\right)P_t(q_{4,1,0,0}) \\
\frac{\mathrm{d}P_t(q_{4,0,1,0})}{dt} = 30k_{yy}P_t(q_{6,0,0,0}) + 2k^d_{xx}P_t(q_{2,1,1,0}) + 4k^d_{yy}P_t(q_{2,0,2,0}) + k^d_{xy}\cdot P_t(q_{2,0,1,1}) - \left(12(k_{xx}+k_{yy}+k_{xy}) + 2k^d_{yy}\right)P_t(q_{4,0,1,0}) \\
\frac{\mathrm{d}P_t(q_{4,0,0,1})}{dt} = 30k_{xy}P_t(q_{6,0,0,0}) + 2k^d_{xx}P_t(q_{2,1,0,1}) + 2k^d_{yy}P_t(q_{2,0,1,1}) + 2k^d_{xy}\cdot P_t(q_{2,0,0,2}) - \left(12(k_{xx}+k_{yy}+k_{xy}) + k^d_{xy}\right)P_t(q_{4,0,0,1}) \\
\frac{\mathrm{d}P_t(q_{2,2,0,0})}{dt} = 12k_{xx}P_t(q_{4,1,0,0}) + 6k^d_{xx}P_t(q_{0,3,0,0}) + 2k^d_{yy}P_t(q_{0,2,1,0}) + k^d_{xy}\cdot P_t(q_{0,2,0,1}) - (2k_{xx}+2k_{yy}+2k_{xy}+4k^d_{xx})P_t(q_{2,2,0,0}) \\
\frac{\mathrm{d}P_t(q_{2,1,1,0})}{dt} = 12k_{xx}P_t(q_{4,0,1,0}) + 12k_{yy}P_t(q_{4,1,0,0}) + 4k^d_{xx}P_t(q_{0,2,1,0}) + 4k^d_{yy}P_t(q_{0,1,2,0}) + k^d_{xy}\cdot P_t(q_{0,1,1,1}) - \left(2k_{xx}+2k_{yy}+2k_{xy}+2k^d_{xx}+2k^d_{yy}\right)P_t(q_{2,1,1,0}) \\
\frac{\mathrm{d}P_t(q_{2,1,0,1})}{dt} = 12k_{xx}P_t(q_{4,0,0,1}) + 12k_{xy}P_t(q_{4,1,0,0}) + 4k^d_{xx}P_t(q_{0,2,0,1}) + 2k^d_{yy}P_t(q_{0,1,1,1}) + 2k^d_{xy}\cdot P_t(q_{0,1,0,2}) - \left(2k_{xx}+2k_{yy}+2k_{xy}+2k^d_{xx}+k^d_{xy}\right)P_t(q_{2,1,0,1}) \\
\frac{\mathrm{d}P_t(q_{2,0,2,0})}{dt} = 12k_{yy}P_t(q_{4,0,1,0}) + 2k^d_{xx}P_t(q_{0,1,2,0}) + 6k^d_{yy}P_t(q_{0,0,3,0}) + k^d_{xy}\cdot P_t(q_{0,0,2,1}) - \left(2k_{xx}+2k_{yy}+2k_{xy}+4k^d_{yy}\right)P_t(q_{2,0,2,0}) \\
\frac{\mathrm{d}P_t(q_{2,0,1,1})}{dt} = 12k_{yy}P_t(q_{4,0,0,1}) + 12k_{xy}P_t(q_{4,0,1,0}) + 4k^d_{xx}P_t(q_{0,1,1,1}) + 4k^d_{yy}P_t(q_{0,0,2,1}) + 2k^d_{xy}\cdot P_t(q_{0,0,1,2}) - \left(2k_{xx}+2k_{yy}+2k_{xy}+2k^d_{yy}+k^d_{xy}\right)P_t(q_{2,0,1,1}) \\
\frac{\mathrm{d}P_t(q_{2,0,0,2})}{dt} = 12k_{xy}P_t(q_{4,0,0,1}) + 2k^d_{xx}P_t(q_{0,1,0,2}) + 2k^d_{yy}P_t(q_{0,0,1,2}) + 3k^d_{xy}\cdot P_t(q_{0,0,0,3}) - \left(2k_{xx}+2k_{yy}+2k_{xy}+2\cdot k^d_{xy}\right)P_t(q_{2,0,0,2}) \\
\frac{\mathrm{d}P_t(q_{0,3,0,0})}{dt} = 2k_{xx}P_t(q_{2,2,0,0}) - 6k^d_{xx}P_t(q_{0,3,0,0}) \\
\frac{\mathrm{d}P_t(q_{0,2,1,0})}{dt} = 2k_{xx}P_t(q_{2,1,1,0}) + 2k_{yy}P_t(q_{2,2,0,0}) - \left(4k^d_{xx}+2k^d_{yy}\right)P_t(q_{0,2,1,0}) \\
\frac{\mathrm{d}P_t(q_{0,2,0,1})}{dt} = 2k_{xx}P_t(q_{2,1,0,1}) + 2k_{xy}P_t(q_{2,2,0,0}) - \left(4k^d_{xx}+k^d_{xy}\right)P_t(q_{0,2,0,1}) \\
\frac{\mathrm{d}P_t(q_{0,1,2,0})}{dt} = 2k_{xx}P_t(q_{2,0,2,0}) + 2k_{yy}P_t(q_{2,1,1,0}) - \left(2k^d_{xx}+4k^d_{yy}\right)P_t(q_{0,1,2,0}) \\
\frac{\mathrm{d}P_t(q_{0,1,1,1})}{dt} = 2k_{xx}P_t(q_{2,0,1,1}) + 2k_{yy}P_t(q_{2,1,0,1}) + 2k_{xy}P_t(q_{2,1,1,0}) - \left(2k^d_{xx}+2k^d_{yy}+k^d_{xy}\right)P_t(q_{0,1,1,1}) \\
\frac{\mathrm{d}P_t(q_{0,1,0,2})}{dt} = 2k_{xx}P_t(q_{2,0,0,2}) + 2k_{xy}P_t(q_{2,1,0,1}) - \left(2k^d_{xx}+2k^d_{xy}\right)P_t(q_{0,1,0,2}) \\
\frac{\mathrm{d}P_t(q_{0,0,3,0})}{dt} = 2k_{yy}P_t(q_{2,0,2,0}) - 6k^d_{yy}P_t(q_{0,0,3,0}) \\
\frac{\mathrm{d}P_t(q_{0,0,2,1})}{dt} = 2k_{yy}P_t(q_{2,0,1,1}) + 2k_{xy}P_t(q_{2,0,2,0}) - \left(4k^d_{yy}+k^d_{xy}\right)P_t(q_{0,0,2,1}) \\
\frac{\mathrm{d}P_t(q_{0,0,1,2})}{dt} = 2k_{yy}P_t(q_{2,0,0,2}) + 2k_{xy}P_t(q_{2,0,1,1}) - \left(2k^d_{yy}+2k^d_{xy}\right)P_t(q_{0,0,1,2}) \\
\frac{\mathrm{d}P_t(q_{0,0,0,3})}{dt} = 2k_{xy}P_t(q_{2,0,0,2}) - 3k^d_{xy}\cdot P_t(q_{0,0,0,3})
\end{cases}
$$

Figure 6.4: Équation maîtresse pour le cas d'étude sur les symétries.

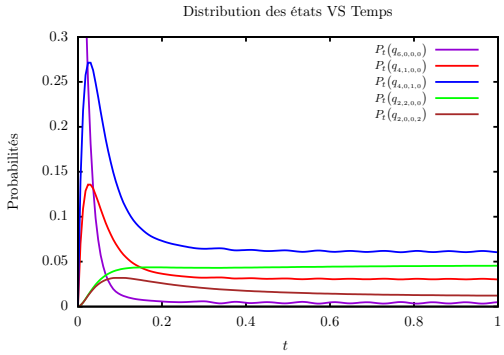


Figure 6.5: Évolution de la distribution de certains états en fonction du temps, avec les paramètres cinétiques $k_{xx} = k_{xy} = 0.5$, $k_{yy} = k^d_{xx} = k^d_{yy} = 1$ et $k^d_{xy} = 4$, et la distribution d'états initiale $P_0\big(q_{6,0,0,0}\big) = 1$.

protéine A, avec tous leurs sites libres. Ainsi, comme les règles d'interaction conservent le nombre d'occurrences des protéines, seuls seront considérés les états i,j,k,l pour lesquels $i + 2 \cdot (j + k + l) = 6$. Le système est dont constitué de vingt variables, en l'occurrence, $q_{6,0,0,0}$, $q_{4,1,0,0}$, $q_{4,0,1,0}$, $q_{4,0,0,1}$, $q_{2,2,0,0}$, $q_{2,1,1,0}$, $q_{2,1,0,1}$, $q_{2,0,2,0}$, $q_{2,0,1,1}$, $q_{2,0,0,2}$, $q_{0,3,0,0}$, $q_{0,2,1,0}$, $q_{0,2,0,1}$, $q_{0,1,2,0}$, $q_{0,1,1,1}$, $q_{0,1,0,2}$, $q_{0,0,3,0}$, $q_{0,0,2,1}$, $q_{0,0,1,2}$ et $q_{0,0,0,3}$.

L'équation maîtresse de notre cas d'étude sur les symétries est donnée en figure 6.4. L'évolution temporelle de la distribution de probabilité de certains états est-elle donnée en figure 6.5 pour les paramètres cinétiques $k_{xx} = k_{xy} = 0.5$, $k_{yy} = k^d_{xx} = k^d_{yy} = 1$ et $k^d_{xy} = 4$.

6.1.1.2.2 Sémantique différentielle. La sémantique différentielle du modèle est définie par le systèmes d'équations différentielles donné en figure 6.6. Ces équations décrivent l'évolution de la quantité des différentes configurations d'espèces biochimique sous les hypothèses de la loi d'action de masse. Les trajectoires de ce système en prenant pour état initial, l'état où les monomères de la protéine A ont pour quantité 6 et les dimères ont pour quantité 0, et pour paramètres cinétiques $k_{xx} = k_{xy} = 0.5$, $k_{yy} = k^d_{xx} = k^d_{yy} = 1$ et $k^d_{xy} = 4$. sont décrites en figure 6.7.

6.1.1.3 Symétries et propriétés comportementales

Sans considérer les aspects quantitatifs, les sites x et y des occurrences de la protéine A sont équivalents. Dès qu'une règle permet de lier le site x d'une occurrence de la protéine A à un site, une autre règle permet de lier le site y de cette occurrence de protéine au même site. De même, quand une règle permet de casser un lien entre le site x d'une occurrence de protéine et un autre site, une autre règle permet de casser ce lien, si ce lien avait été porté par le site y de cette occurrence de protéine.

Il est donc légitime de se poser la question suivante : quelles conséquences les équivalences entre sites peuvent-elles impliquer en terme de comportement du système sous-jacent et pour quelles conditions supplémentaires sur les paramètres cinétiques du modèle et son état initial (ou sa distribution d'états initiale) ?

6.1.2 Modèle simplifié

Il semble naturel de vouloir oublier la différence entre des sites équivalents. Dans notre modèle, cela revient à considérer que les sites x et y sont deux occurrences d'un même site dans les occurrences de la protéine A. À ce niveau d'abstraction, les occurrences de la protéine A auront donc chacune deux sites d'interaction, désormais supposés indiscernables.

6.1.2.1 Configurations d'espèces biochimiques et règles d'interaction.

Comme dessiné en figure 6.8, il ne reste alors que deux sortes d'occurrences de configurations d'espèces biochimiques, les occurrences du monomère de la protéine A (qui ne sont formés que d'une occurrence de la protéine A avec ses deux sites de liaison libres) et les occurrences de dimères de la protéine A (qui sont constitués de deux occurrences de la protéine A liées exactement par une liaison entre un site de l'une et un site de l'autre).

$$\begin{cases} \frac{\mathrm{d}[A]}{dt} = 2k^d_{xy}\,[A.x - y.A] + 2k^d_{yy}2\,[A.y - y.A] + 2k^d_{xx}2\,[A.x - x.A] - 2\,(k_{yy} + k_{xx} + k_{xy})\,[A]^2 \\ \frac{\mathrm{d}[A.x-x.A]}{dt} = k_{xx}\,[A]^2 - k^d_{xx}2\,[A.x - x.A] \\ \frac{\mathrm{d}[A.y-y.A]}{dt} = k_{yy}\,[A]^2 - k^d_{yy}2\,[A.y - y.A] \\ \frac{\mathrm{d}[A.x-y.A]}{dt} = k_{xy}\,[A]^2 - k^d_{xy}\,[A.x - y.A] \end{cases}$$

Figure 6.6: Système d'équations différentielles. La solution de ce système décrit l'évolution des quantités des différentes configurations des espèces biochimiques en fonction du temps, sous les hypothèses de la loi d'Action de Masse. Les coefficients avant les constantes d'interaction représentent le nombre d'unité consommée par réaction, alors que ceux après les constantes d'interaction représentent le nombre de manière d'appliquée la règle correspondante.

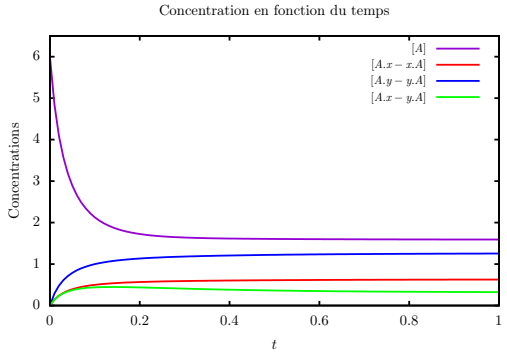


Figure 6.7: Évolution de la quantité des différentes configurations des espèces biochimiques en fonction du temps, avec les paramètres cinétiques $k_{xx} = k_{xy} = 0.5$, $k_{yy} = k^d_{xx} = k^d_{yy} = 1$ et $k^d_{xy} = 4$, et l'état initiat $[A] = 6$ et $[A.x - x.A] = [A.x - y.A] = [A.y - y.A] = 0$.

Par ailleurs, les règles se simplifient de la même manière. Puisqu'il n'y a plus de distinctions entre les sites de liaison de chaque occurrence de la protéine A, les trois règles de liaison peuvent se résumer en une seule. Il en est de même pour les trois règles de dissociation. Les deux règles du modèle simplifié sont données en figure 6.9. Dans le modèle simplifié, la constante de réaction pour la règle de liaison est notée K, alors que celle pour la règle de dissociation K^d.

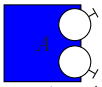
6.1.2.2 États des systèmes stochastiques et différentiels sous-jacent.

Les états du système stochastique sous-jacent sont donc décrits par le nombre d'occurrences du monomère de la protéine A (qui sont formées d'une occurrence de la protéine A) et le nombre d'occurrences du dimère de la protéine A (qui sont formées de deux occurrences de la protéine A liées entre-elles). Un tel état sera noté $Q_{i,j}$ avec i le nombre d'occurrences du monomère et j le nombre d'occurrences de dimères de la protéine A.

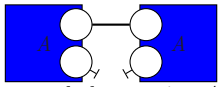
Quant à eux, les états du système différentiel sous-jacent sont décrits par deux quantités. La quantité en monomère est notée $[A_1]$ alors que celle en dimère est notée $[A_2]$.

6.1.2.3 Systèmes dynamiques sous-jacent

6.1.2.3.1 Équation maîtresse. L'équation maîtresse du modèle simplifié est donnée en figure 6.10. La solution de ce système d'équations définit donc l'évolution de la distribution de probabilités du nombre de dimères au cours du temps dans la sémantique stochastique. Seuls les états accessibles à partir de six monomères ont été considérés, c'est à dire les quatre états potentiels $Q_{6,0}$, $Q_{4,1}$, $Q_{2,2}$ et $Q_{0,3}$.

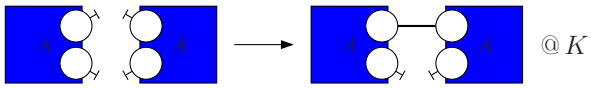


(a) Une occurrence de la protéine A avec ses deux sites libres.

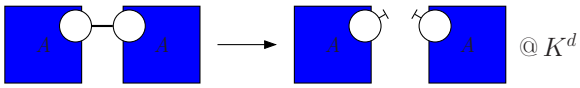


(b) Deux occurrences de la protéine A liées entre elles.

Figure 6.8: Les deux configurations des espèces biochimiques du modèle simplifié. Le nom des sites a été retiré car ils sont maintenant supposés indiscernables.



(a) La règle de liaison.



(b) La règle de dissociation.

Figure 6.9: Les deux règles du modèle simplifié. En 6.9(a), deux occurrences de la protéine A peuvent se lier avec une constante de réaction K (peu importe par quels sites) quand tous leurs sites sont libres. En 6.9(b), le lien entre deux occurrences de la protéine A peut se rompre avec une constante de réaction K^d.

6.1.2.3.2 Sémantique différentielle. La sémantique différentielle est quant à elle donnée en figure 6.11. Elle donne l'évolution de la quantité en monomère, $[A_1]$, et en dimère, $[A_2]$, au cours du temps, sous les hypothèses de la loi d'action de masse.

6.1.3 Comparaison des dynamiques des deux modèles

6.1.3.1 Quotient

Les composants de modèle simplifié ont été définis de manière intentionnelle sous la forme de graphes à sites. Quelques écarts ont été pris par rapport à la syntaxe de Kappa. En effet, celle-ci ne permet pas de répéter des sites dans l'interface d'une protéine. Cette description intentionnelle permet des raisonnements intuitifs sur le comportement des composants du modèle simplifié. Toutefois, une caractérisation extensionnelle de ces composants est requise pour relier formellement les composants du modèle simplifié au composants du modèle initial et ainsi justifier rigoureusement ces raisonnements. Il suffit pour cela d'interpréter les configurations d'espèces biochimiques et les états du modèle simplifié respectivement comme des classes d'équivalences de configurations d'espèces biochimiques et d'états du modèle initial. Les configurations d'espèces biochimiques sont regroupées en deux classes, les monomères et les dimères. Ainsi, dans le cadre différentiel, les variables du modèle simplifié correspondent à la somme des quantités des configurations d'espèces biochimiques dans chaque classe d'équivalences, comme indiqué en figure 6.13. De la même manière, les états du système stochastique initial peuvent être regroupés selon le nombre d'occurrences du monomère de la protéine A et le nombre global des trois types de dimères de la protéine A. Du coup, à partir d'un état initial formé de 6 occurrences de la protéine A aves ses deux sites sont libres, il est possible d'attendre des états dans exactement 4 classes d'équivalence, selon qu'il y ait 0, 1, 2 ou 3 dimères. Ceci peut être étendu aux distributions d'états qui apparaissent dans l'équation maîtresse : la probabilité d'avoir exactement i monomères et j dimères est alors vue comme la somme des probabilité des états du modèle initial ayant exactement i monomères et j dimères. Cette relation est exprimée en figure 6.12 pour les états accessibles à partir de 6 occurrences de la protéine A avec les deux sites de liaison libres.

Il existe donc deux manières de calculer le comportement du modèle simplifié : soit en résolvant directement les équations du modèle simplifié, soit en résolvant le modèle initial et en calculant le comportement du modèle simplifié à l'aide de la caractérisation extensionnelle de ses composants.

En figure 6.14 est montrée la comparaison empirique entre ces deux méthodes pour plusieurs jeux de paramètres, que ce soit dans la cadre stochastique ou différentiel. La colonne de gauche est consacrée aux solutions des équations maîtresses, alors que la colonne de droite porte sur celles des équations différentielles sur les quantités de chaque composant. Quatre jeux de paramètres sont considérés. Même si les paramètres n'ont pas la même signification dans le cadre stochastique et dans celui différentiel, la comparaison est faîte avec les mêmes valeurs numériques. Dans chaque graphique, les lignes continues sont obtenues en considérant le modèle initial et en projetant les résultats à l'aide de la caractérisation extensionnelle des composants du modèle simplifié, alors que les courbes obtenues directement avec le modèle simplifié sont dessinées avec des

$$\begin{cases} \frac{\mathrm{d}P_t(Q_{6,0})}{dt} = 2K^d P_t(Q_{4,1}) - 30K P_t(Q_{6,0}) \\ \frac{\mathrm{d}P_t(Q_{4,1})}{dt} = 30K P_t(Q_{6,0}) + 4K^d P_t(Q_{2,2}) - \left(12K + 2K^d\right) P_t(Q_{4,1}) \\ \frac{\mathrm{d}P_t(Q_{2,2})}{dt} = 12K P_t(Q_{4,1}) + 6K^d P_t(Q_{0,3}) - \left(2K + 4K^d\right) P_t(Q_{2,2}) \\ \frac{\mathrm{d}P_t(Q_{0,3})}{dt} = 2K P_t(Q_{2,2}) - 6K^d P_t(Q_{0,3}) \end{cases}$$

Figure 6.10: Équation maîtresse pour le modèle simplifié.

$$\begin{cases} \frac{\mathrm{d}[A_1]}{dt} = 2K^d 2\,[A_2] - 2K\,[A_1]^2 \\ \frac{\mathrm{d}[A_2]}{dt} = K\,[A_1]^2 - 2K^d\,[A_2] \end{cases}$$

Figure 6.11: Système différentiel pour le modèle simplifié. Les coefficients avant les constantes d'interaction représentent le nombre d'unité consommée par réaction, alors que ceux après les constantes d'interaction représentent le nombre de manière d'appliquée la règle correspondante.

tirets.

Le but de ces simulations numériques est de tester l'importance de trois contraintes. La première contrainte concerne les constantes de réaction pour les règles de dissociation. La seconde concerne l'état (ou la distribution d'états) initial(le). Enfin, la troisième concerne les constantes de réaction pour les règles de liaison.

- **Contrainte sur les constantes de dissociation.** Intuitivement, pour que les sites x et y des occurrences de la protéine A tiennent un rôle symétrique, il est nécessaire que chaque lien puisse se défaire avec la même vitesse. Notons que les règles de dissociation pour les liens symétriques (entre deux sites x ou entre deux sites y) peuvent s'appliquer deux fois. Aussi pour chaque type de lien puisse se défaire avec la même cinétique, il faut que les constantes k^d_{xx} et k^d_{yy} soient toutes deux égales à la moitié de la constante k^d_{xy}. Enfin, pour que le modèle simplifié puisse simuler ce comportement, il faut prendre la constante K^d égale à la valeur commune des deux constantes k^d_{xx} et k^d_{yy}.

- **Contrainte sur l'état initial ou la distribution initiale.** La seconde contrainte concerne l'état initial (ou la distribution initiale des états). Elle a pour but de tester empiriquement l'importance que les sites x et y soient, ou non, indiscernables dans la distribution initiale (ou dans l'état initial). Pour cela, sous cette hypothèse, on ne considérera que des occurrences de la protéine A dont tous les sites x et y sont libres (puisque c'est la seule configuration de la protéine A dans laquelle les deux sites sont dans le même état).

- **Contrainte sur les constantes de liaison.** La troisième contrainte concerne les règles de liaison. On peut se demander s'il est important que les liaisons entre deux occurrences libres de la protéine A peuvent s'établir de manière équiprobable entre les 4 positions potentielles (x-x, y-y, x-y et y-x, on a ici distingué les deux occurrences de la protéine A dans les dimères). Les règles de liaison symétrique offrent chacune deux manières de créer la liaison correspondante (car les deux occurrences de la protéine A jouent exactement le même rôle dans la règle), alors que la règle de liaison asymétrique peut établir deux liaisons différentes si l'on distingue les deux occurrences de la protéine A auxquelles on applique la règle de liaison. De ce fait, la troisième contrainte spécifie que les constantes k_{xx} et k_{yy} sont égales et que la constante k_{xy} est égale au double de la contrainte k_{xx}. Par ailleurs, pour simuler ces règles, la constante K sera choisie comme la somme des trois constantes k_{xx}, k_{yy} et k_{xy}. En effet, les trois règles de liaison du modèle initial, ainsi que celle du modèle simplifié, peuvent s'appliquer dans deux positions chacune par paire d'occurrences libres de la protéine A.

La correspondance entre le modèle initial et le modèle simplifiée est testée, pour la sémantique stochastique et pour la sémantique différentielle, sous quatre scénarii. Dans le premier, les trois contraintes de symétrie sont satisfaites, alors que dans les autres exactement une de ces contraintes est invalidée.

La description de chaque scénario et des résultats obtenus est donnée ci-dessous :

$$\begin{cases} P_t(Q_{6,0}) = P_t\big(q_{6,0,0,0}\big) \\ P_t(Q_{4,1}) = P_t\big(q_{4,1,0,0}\big) + P_t\big(q_{4,0,1,0}\big) + P_t\big(q_{4,0,0,1}\big) \\ P_t(Q_{2,2}) = P_t\big(q_{2,2,0,0}\big) + P_t\big(q_{2,1,1,0}\big) + P_t\big(q_{2,1,0,1}\big) + P_t\big(q_{2,0,2,0}\big) + P_t\big(q_{2,0,1,1}\big) + P_t\big(q_{2,0,0,2}\big) \\ P(Q_{0,3}) = P_t\big(q_{0,3,0,0}\big) + P_t\big(q_{0,2,1,0}\big) + P_t\big(q_{0,2,0,1}\big) + P_t\big(q_{0,1,2,0}\big) + P_t\big(q_{0,1,1,1}\big) + P_t\big(q_{0,1,0,2}\big) \\ \qquad\qquad + P_t\big(q_{0,0,3,0}\big) + P_t\big(q_{0,0,2,1}\big) + P_t\big(q_{0,0,0,3}\big) + P_t\big(q_{0,0,1,2}\big) + P_t\big(q_{0,1,0,2}\big). \end{cases}$$

Figure 6.12: Définition extensionnelle des variables de l'équation maîtresse du modèle simplifié.

$$\begin{cases} [A_1] = [A] \\ [A_2] = [A.x - x.A] + [A.y - y.A] + [A.x - y.A] \end{cases}$$

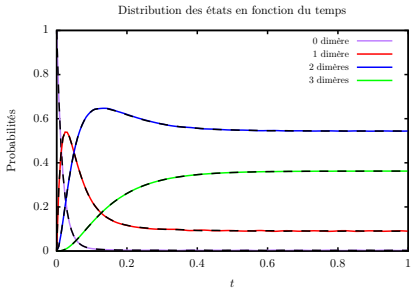
Figure 6.13: Définition extensionnelle des variables du modèle simplifié pour le cadre différentiel.

1. Dans le premier jeu de paramètres, les trois contraintes sont satisfaites. Les constantes k_{xx} et k_{yy} sont fixées à $\frac{1}{2}$, les constantes k_{xy}, k^d_{xx} et k^d_{yy} sont fixées à 1 alors que la constante k^d_{xy} ext fixée à 2. Le système stochastique sous-jacent débute de l'état formé de 6 occurrences de la protéine A sans lien, et le système différentiel de l'état où les monomères sont présents en quantité 6, et les dimères absents. Les courbes montrent que la distribution des états du modèle simplifié correspond exactement à celle du modèles initials (voir en figure 6.14(a)) : les mêmes valeurs sont obtenues en calculant l'évolution de la distribution des états dans le modèle initial et en regroupant les états selon le nombre d'occurrences de dimère (courbes continues) ou en calculant la distribution des états dans le modèle simplifié directement (tirets). Le même phénomène se retrouve dans les systèmes différentiels sous-jacents. Les courbes en figure 6.14(b) montrent que les mêmes quantités sont obtenues en regroupant la quantité de dimères dans le modèle initial ou en calculant directement cette quantité dans le modèle simplifié.

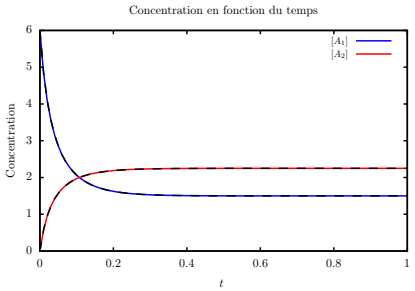
2. Dans le second jeu de paramètres, la contrainte sur les constantes de dissociation n'est plus satisfaite. Les constantes d'association et la distribution initiale des états (ou l'état initial dans le cadre différentiel) sont gardées telles quelles, ainsi que les constantes de dissociation k^d_{xx} et k^d_{yy}. Mais la constante de dissociation k^d_{xy} est maintenant fixée à 4. La constante de dissociation k^d_{xy} n'étant pas égale au double de la constante de dissociation k^d_{xx}, il n'y a pas de manière intuitive de fixer la constante de dissociation du modèle simplifié. Elle est prise arbitrairement égale à la moyenne entre la constante de dissociation des liens symétriques et la moitiée celle des liens asymétriques, à savoir 3. On peut alors constater à la fois dans le cadre stochastique et dans le cadre différentiel un écart entre le comportement du modèle initial et du modèle simplifié (voir en figure 6.14(c) et en figure 6.14(d)). Une telle disparité aurait été observée quelque soit la valeur de la constante K^d.

3. Le troisième jeu de paramètres a pour but de tester l'importance éventuelle de l'état initial. Par rapport au premier jeu de paramètres, seule la distribution initiale des états (ou l'état initial dans le cadre différentiel) est modifiée. Le système stochastique débute dans un état avec deux occurrences de la protéine A sans lien, et deux occurrences du dimère symétrique de la protéine A formées par des liens sur les sites x. Quant au système différentiel, il démarre d'un état où les monomères et les dimères $x - x$ sont en quantité 2, alors que les deux autres formes de dimère sont en quantité 0. l'évolution des distributions d'états des deux systèmes stochastiques sous-jacents est dessinée en figure 6.14(e), alors que l'évolution des quantités de monomères et de dimères dans les deux systèmes différentiels sous-jacents est donnée en figure 6.14(f). À la fois, le cadre stochastique et dans le cadre différentiel, les courbes coïncident entre le modèle initial et le modèle simplifié, ce qui suggère que la correction de la simplification du modèle ne dépend pas de la distribution initiale dans le cadre stochastique ou de l'état initial dans le cadre différentiel.

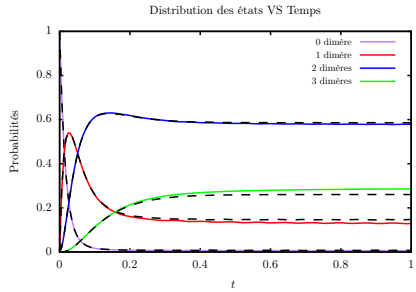
4. Enfin, le quatrième jeu de paramètres vise à tester l'impact de la contrainte sur les constantes d'association. Les paramètres sont les mêmes que dans le premier jeu, sauf la constante k_{yy} qui est fixée à 1 et la constante k_{xy} qui est fixée à 2. De plus, la constante d'association dans le modèle simplifié, K, reste la somme des constantes d'association dans le modèle initial, c'est à dire $\frac{7}{2}$. Là encore, le comportement du modèle initial et celui du modèle simplifié coincident, que ce soit pour la sémantique stochastique ou la sémantique différentielle (voir en figure 6.14(g) et en figure 6.14(h)).



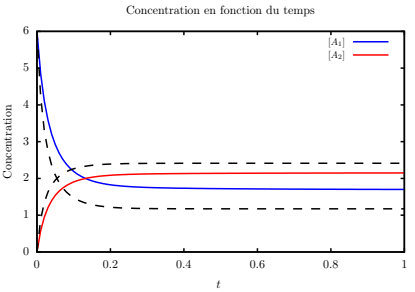
(a) $k_{xx} = k_{yy} = 0.5$, $k_{xy} = k_{xx}^d = k_{yy}^d = 1$, $k_{xy}^d = 2$ et $P_0\big(q_{6,0,0,0,0}\big) = 1$; $K = 2$, $K^d = 1$ et $P_0(Q_{6,0}) = 1$.



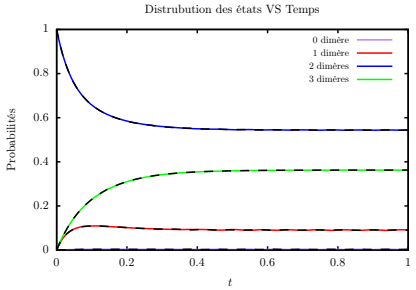
(b) $k_{xx} = k_{yy} = 0.5$, $k_{xy} = k_{xx}^d = k_{yy}^d = 1$, $k_{xy}^d = 2$, initialement $[A] = 6$ et $[A.x - x.A] = [A.x - y.A] = [A.y - y.A] = 0$; $K = 2$, $K^d = 1$, initialement $[A_1] = 6$ et $[A_2] = 0$.



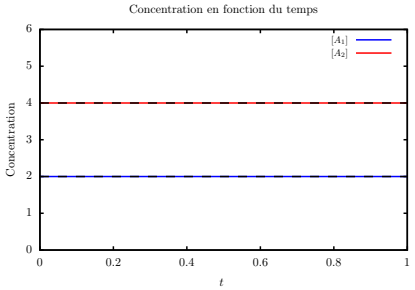
(c) $k_{xx} = k_{yy} = 0.5$, $k_{xy} = k_{xx}^d = k_{yy}^d = 1$, $k_{xy}^d = 4$ et $P_0\big(q_{6,0,0,0,0}\big) = 1$; $K = 2$, $K^d = 1.5$ et $P_0(Q_{6,0}) = 1$.



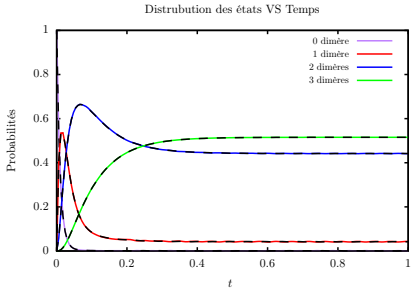
(d) $k_{xx} = k_{yy} = 0.5$, $k_{xy} = k_{xx}^d = k_{yy}^d = 1$, $k_{xy}^d = 4$, initialement $[A] = 6$ sans dimère ; $K = 2$, $K^d = 1.5$, initialement $[A_1] = 6$ et $[A_2] = 0$.



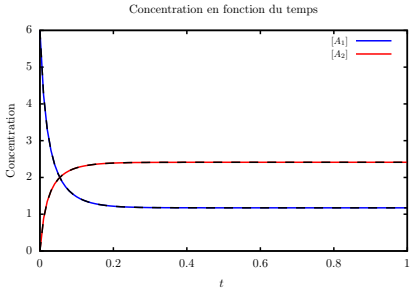
(e) $k_{xx} = k_{yy} = 0.5$, $k_{xy} = k_{xx}^d = k_{yy}^d = 1$, $k_{xy}^d = 2$ et $P_0\big(q_{2,2,0,0,0}\big) = 1$; $K = 2$, $K^d = 1$, $P_0(Q_{2,2}) = 1$.



(f) $k_{xx} = k_{yy} = 0.5$, $k_{xy} = k_{xx}^d = k_{yy}^d = 1$, $k_{xy}^d = 2$, initialement $[A] = [A.x - x.A] = 2$ et $[A.y - y.A] = [A.x - y.A] = 0$; $K = 2$, $K^d = 1$, initialement $[A_1] = [A_2] = 2$.



(g) $k_{xx} = 0.5$, $k_{yy} = 1$, $k_{xy} = 2$, $k_{xx}^d = k_{yy}^d = 1$, $k_{xy}^d = 2$ et $P_0\big(q_{6,0,0,0,0}\big) = 1$; $K = 3.5$, $K^d = 1$ et $P_0(Q_{6,0}) = 1$.



(h) $k_{xx} = 0.5$, $k_{yy} = 1$, $k_{xy} = 2$, $k_{xx}^d = k_{yy}^d = 1$, $k_{xy}^d = 2$, initialement $[A] = 6$ sans dimère ; $K = 3.5$, $K^d = 1$, initialement $[A_1] = 6$ et $[A_2] = 0$.

Figure 6.14: Comparaison entre le comportement du modèle initial (traits continus) et celui du modèle simplifié (tirets). À gauche, évolution de la distribution de probabilités du nombre de dimères pour différents paramètres cinétiques et différentes distributions initiales des états dans le système stochastique sous-jacent. À droite, évolution de la quantité en dimère pour différents paramètres cinétiques et différentes quantités initiales.

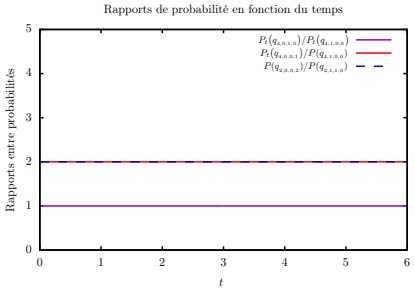
Ainsi, si la contrainte sur les constantes de dissociation semble cruciale pour pouvoir simplifier le modèle de manière exacte, les choix de la distribution initiale (ou de l'état initial) et des constantes d'association ne semblent pas importants. Pour les constantes d'association, ceci s'explique par la simplicité du cas d'étude. En effet, les règles de liaison ne lient que des monomères, or ces monomères sont invariants par échange de leurs sites x et y. De manière général, les conditions sur les constantes cinétique portent sur des règles qui effectuent des actions similaires sur des motifs équivalents à échange de sites près. La correction du modèle simplifié peut alors s'expliquer par une bisimulation en-avant [22] sur l'espace des états du système stochastique sous-jacent ou de son analogue différentiel [34].

6.1.3.2 Invariants quantitatifs

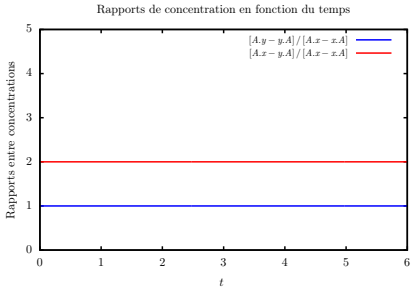
Dans le paragraphe 6.1.2, le fait que certains sites puissent partager exactement les mêmes capacités d'interaction a été exploité pour simplifier un modèle jouet de manière exacte. Les composants du modèle simplifié ont été reliés formellement à ceux du modèle initial et les mêmes résultats ont été obtenus en exécutant le modèle initial avant d'utiliser ces relations formelles pour en déduire l'évolution des composants du modèle simplifié ou en exécutant directement le modèle simplifié.

Une autre classe de propriétés intéressantes est celle des invariants quantitatifs. Il semble naturel de supposer que lorsque deux sites sont équivalents dans toutes les occurrences d'une protéine, alors, dans le cadre stochastique, les états obtenus en permutant ces deux sites dans une ou plusieurs occurrences de cette protéine dans un état donné soient équiprobables. De même, dans le cadre différentiel, les quantités des configurations d'espèces biochimiques obtenues en échangeant ces deux sites dans une ou plusieurs occurrences de cette protéine devraient être égales. En fait, ces quantités ne le sont pas en pratique, elles sont plutôt proportionnelles. Pour mieux comprendre ce phénomène, nous pouvons étudier ce qui se passe dans un jeu de "pile" ou "face" avec deux pièces. Lorsque les deux pièces sont jetées, elles tombent toutes deux sur le côté "pile" avec une probabilité un quart, toutes deux sur le côté "face" avec une probabilité un quart et l'une sur le côté "pile" et l'autre sur le côté "face" avec une probabilité un demi. Pourtant il est possible d'intervertir les côtés "pile" et "face" d'une ou des deux pièces sans changer la nature du jeu. Les côtés "pile" et "face" ont donc un rôle symétrique. Pourtant les trois configurations ne sont pas équiprobables, en fait leur probabilité est proportionnelle au nombre de manière de retourner les pièces sans changer la configuration globale. Ainsi, la configuration comportant les deux côtés "pile" et "face" reste inchangée lorsque les deux pièces sont retournées simultanément (c'est la seule manière non triviale de laisser cette configuration inchangée), alors qu'un tirage double est changé dès que l'on retourne au moins une des deux pièces, d'où un rapport de 2 sur 1 (en tenant compte de la transformation triviale qui consiste à ne changer aucune pièce). Ainsi, il faut tenir compte du nombre des transformations qui laissent un état ou une configuration d'espèce biochimique inchangé pour connaître les rapports de proportionnalité entre les probabilités d'être dans deux états symétriques ou entre les quantités de deux configurations symétriques d'espèces biochimiques, or ce nombre est inversement proportionnel au nombre d'automorphismes dans ces configurations.

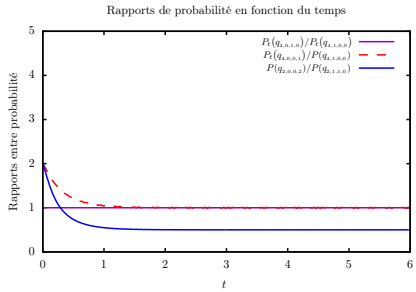
Nous étudions maintenant l'importance des trois contraintes qui avaient été utilisées pour tester l'adéquation entre le modèle initial et le modèle simplifié. Ici, ces conditions doivent assurer, dans le cadre stochastique, que les probabilités d'être dans deux états obtenus en remplaçant une occurrence d'un type de dimère par une occurrence d'un autre type restent dans le même rapport de proportionnalité au cours du temps. Dans le cadre différentiel, ces conditions doivent assurer que les quantités entre les différentes sortes de dimères restent dans le même rapport de proportionnalité aux cours du temps. En figure 6.15 est vérifié de manière empirique si de tels rapports de proportionnalité se manifestent pour plusieurs jeux de paramètres, que ce soit dans le cadre stochastique (colonne de gauche) ou différentiel (colonne de droite), en faisant varier la valeur des constantes de dissociations, la distribution initiale (ou l'état initial), et les constantes d'association. Dans le cadre stochastique, l'évolution de trois rapports est considéré : le rapport entre l'état composé de quatre occurrences du monomère de la protéine A et d'une occurrence du dimère symétrique x-x de la protéine A et celui comprenant quatre occurrences du monomère de la protéine A et une occurrence du dimère symétrique y-y ; le rapport entre l'état composé de quatre occurrences du monomère de la protéine A et d'une occurrence du dimère symétrique x-x de la protéine A et celui comportant quatre occurrences du monomère de la protéine A et une occurrence du dimère asymétrique de la protéine A et le rapport entre l'état composé de deux occurrences du monomère de la protéine A et de deux occurrences du dimère asymétrique de la protéine A et celui formé de deux occurrences du monomère de la protéine A et d'une occurrence de chacune des sortes de dimères symétriques de la protéine A. Dans le cadre différentiel, est considérée l'évolution des rapports entre les quantités des différentes sortes des dimères de la protéine A.



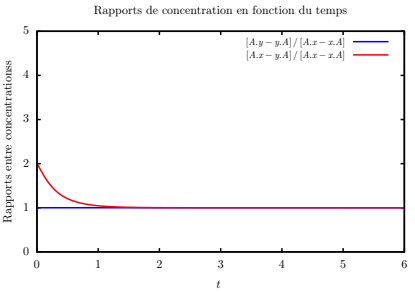
(a) $k_{xx} = k_{yy} = 0.5$, $k_{xy} = k^d_{xx} = k^d_{yy} = 1$, $k^d_{xy} = 2$ et $P_0(q_{6,0,0,0}) = 1$.



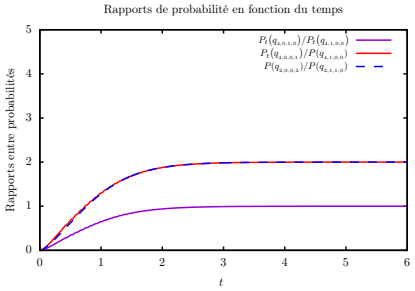
(b) $k_{xx} = k_{yy} = 0.5$, $k_{xy} = k^d_{xx} = k^d_{yy} = 1$ $k^d_{xy} = 2$, initialement $[A] = 6$ et $[A.x - x.A] = [A.x - y.A] = [A.y - y.A] = 0$.



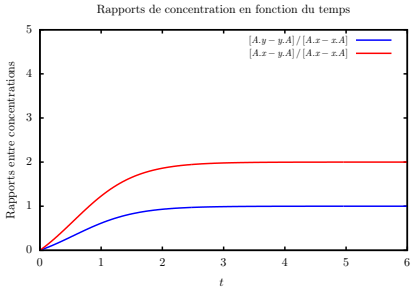
(c) $k_{xx} = k_{yy} = 0.5$, $k_{xy} = k^d_{xx} = k^d_{yy} = 1$, $k^d_{xy} = 4$ et $P_0(q_{6,0,0,0}) = 1$.



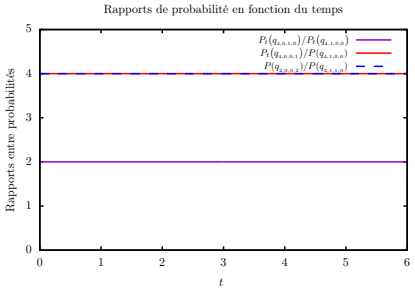
(d) $k_{xx} = k_{yy} = 0.5$, $k_{xy} = k^d_{xx} = k^d_{yy} = 1$, $k^d_{xy} = 4$, initialement $[A] = 6$ sans dimère.



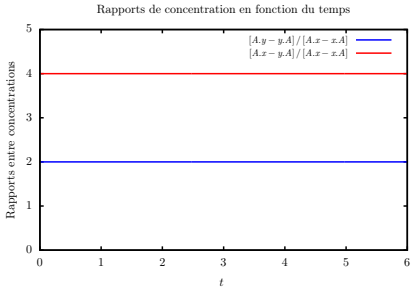
(e) $k_{xx} = k_{yy} = 0.5$, $k_{xy} = k^d_{xx} = k^d_{yy} = 1$, $k^d_{xy} = 2$ et $P_0(q_{2,2,0,0}) = 1$.



(f) $k_{xx} = k_{yy} = 0.5$, $k_{xy} = k^d_{xx} = k^d_{yy} = 1$, $k^d_{xy} = 2$, initialement $[A] = [A.x - x.A] = 2$ et $[A.y - y.A] = [A.x - y.A] = 0$.



(g) $k_{xx} = 0.5$, $k_{yy} = 1$, $k_{xy} = 2$, $k^d_{xx} = k^d_{yy} = 1$, $k^d_{xy} = 2$ et $P_0(q_{6,0,0,0}) = 1$.



(h) $k_{xx} = 0.5$, $k_{yy} = 1$, $k_{xy} = 2$, $k^d_{xx} = k^d_{yy} = 1$, $k^d_{xy} = 2$, initialement $[A] = 6$ sans dimère.

Figure 6.15: Relations de proportionnalité dans le modèle initial. À gauche, évolution de rapports entre la probabilité d'être dans certains états pour différents paramètres cinétiques et différentes distributions initiales des états dans le système stochastique sous-jacent. À droite, évolution de rapports entre les quantités de différentes configurations des espèces biochimiques pour différents paramètres cinétiques et différentes quantités initiales.

1. Dans le premier jeu de paramètres, les trois contraintes sont satisfaites. Les constantes k_{xx} et k_{yy} sont fixées à 0.5, les constantes k_{xy}, k^d_{xx} et k^d_{yy} sont fixées à 1, et la constante k^d_{xy} est fixée à 2. Le système stochastique sous-jacent débute de l'état formé de 6 occurrences de la protéine A sans lien et le système différentiel de l'état où les monomères sont présents en quantité 6 et les dimères absents. En figure 6.15(a), les courbes montrent de manière empirique que la probabilité d'être dans l'état avec quatre occurrences de la protéine A sous forme de monomère et deux occurrences de la protéine A liées par leurs sites x est toujours la même que celle d'être dans l'état avec quatre occurrences de la protéine A sous forme de monomère et deux occurrences de la protéine A liées par leurs sites y. Par ailleurs, cette probabilité est toujours la moitié de celle d'être dans l'état avec quatre occurrences de la protéine A sous forme de monomère et deux occurrences de la protéine A liées par le site x de l'un et le site y de l'autre. Ces rapports de proportionnalité correspondent à ceux déjà observés dans le jeu de "pile" ou "face". Enfin, la probabilité d'être dans un état avec deux occurrences de la protéine A sous forme de monomère et deux occurrences de dimères asymétriques est toujours deux fois plus grande que celle d'être dans un état avec deux occurrences de la protéine A sous forme de monomère et d'une occurrence de chaque forme de dimère symétrique. Ce rapport s'explique par le fait que la formation d'un dimère asymétrique est deux fois plus probable que celle d'un type donnée de dimère symétrique (d'où un facteur 4). Cependant, il faut diviser ce facteur par 2, car deux dimères symétriques différents peuvent être obtenu en prenant le premier formé par une liaison entre deux sites x et le second entre deux sites y, ou l'inverse, soit deux fois plus de possibilités.

   En ce qui concerne le comportement de la solution du système différentiel sous-jacent, les courbes en figure 6.15(b) montrent que les quantités des deux configurations symétries du dimère restent toujours égales, alors que la quantité de la configuration asymétrique du dimère est toujours égale au double de cette valeur commune.

2. Dans le second jeu de paramètres, la contrainte sur les constantes de dissociation est relâchée. Les constantes de dissociations sont ainsi fixées de manière arbitraire à $k^d_{xx} = 1$, $k^d_{yy} = 1$ et $k^d_{xy} = 4$, alors que les autres paramètres restent les mêmes que pour le premier jeu de paramètres. Les courbes dessinées en figure 6.15(c) montrent que la probabilité d'être dans l'état avec quatre occurrences de la protéine A sous forme de monomère et deux occurrences de la protéine A liées par leurs sites x est toujours la même que celle d'être dans l'état avec quatre occurrences de la protéine A sous forme de monomère et deux occurrences de la protéine A liées par leurs sites y. Par contre, les autres probabilités étudiées ne sont pas proportionnelles. Initialement, la probabilité d'être dans un état avec une occurrence du dimère asymétrique et quatre occurrences de la protéine A sous la forme de monomère est deux fois plus grande que celle d'être dans un état avec une occurrence du dimère asymétrique et quatre occurrences de la protéine A sous la forme de monomère, mais ce rapport tends vers un avec le temps. D'autre part, la probabilité d'être dans un état avec deux occurrences du dimère asymétrique et deux occurrences de la protéine A sous la forme de monomère est deux fois plus grande au début que la probabilité d'être dans un état avec une occurrence de chacun des types de dimères symétriques et deux occurrences de la protéine A sous la forme de monomère, mais ce rapport tends vers un demi avec le temps. Dans le système différentiel, les quantités des deux configurations symétriques du dimère sont toujours égales, alors que la quantité de la configuration asymétrique du dimère est initialement deux fois plus grande, pour finalement converger vers cette même valeur. En fait, les rapports initiaux sont dictés par les constantes d'association, ce sont les mêmes que pour le premier jeu de paramètres. En revanche, les rapports à la limite sont dictés par le rapport entre les constantes respectives d'association et de dissociation (et la répétition des occurrences du dimère asymétrique pour le facteur un demi additionnel).

3. Dans le troisième jeu de paramètres, c'est la distribution d'états initiale (dans le cadre stochastique) et l'état initial (dans le cadre différentiel) qui sont fixés arbitrairement. Il n'y a de rapports de proportionnalité ni dans le cadre stochastique (voir en figure 6.15(e)), ni dans le cadre différentiel (voir en figure 6.15(f)). Initialement, les rapports de probabilités et de quantités sont imposés par les abondances au début de l'exécution des deux systèmes. À la limite, l'impact du choix de la distribution initiale ou de l'état initial tend à disparaître et les rapports trouvés avec le premier jeu de paramètres resurgissent.

4. Enfin, le quatrième jeu de paramètres est obtenu en fixant de manière arbitraire les constantes d'association. Ceci a pour effet de déplacer les rapports de probabilités et les rapports de quantités, qui sont maintenant fixé par les rapports entre les constantes de liaison et celles de dissociation(voir en figure 6.14(g) et en figure 6.14(h)).

L'invariance de ces rapports de proportionnalité, pour certains jeux de paramètres est révélateur de la présence d'une relation de groupage faible. Celle-ci permet de quotienter l'exécution d'une chaîne de Markov pour certaines distributions d'états initiales tout en restant Markovien [21]. De même, cela permet de réduire le système différentiel sous-jacent pour certains états initiaux.

Dans le cas du premier jeu de paramètres, le quotient opère même à plus bas-niveau, puisqu'il s'exprime au niveau réactionnel: à chaque réaction peut être associée une réaction symétrique obtenue en échangeant les occurrences des sites x et y dans certaines occurrences de la protéine A, ce qui permet de montrer l'existence d'une bisimulation arrière [22] sur système de transition stochastique sous-jacent ou de réduire le système différentiel sous-jacent.

6.1.4 Conclusion sur le cas d'étude

Pour conclure avec cet exemple jouet, nous avons vu que des symétries pouvaient apparaître dans les configurations d'espèces biochimiques, dans les états du système, dans les distributions d'états ou même dans l'effet des règles. Ces symétries ont un impact sur notre capacité à réduire la dynamique du système. En effet, lorsque l'action des règles est symétrique, il est possible d'induire une relation d'équivalence pour quotienter l'ensemble des états, ce qui suggère l'existence d'une bisimulation avant. De plus, lorsque la distribution initiale des états est également symétrique, alors le système non simplifié comporte des invariants statistiques, ce qui suggère l'existence d'une bisimulation arrière. Contrairement aux raisonnements sur les rapport entre constantes de liaison et de dissociation, c'est un raisonnement compositionnel qui considère les réactions, classe de symétries par classe de symétries. C'est pourquoi ce type de raisonnement sera privilégié dans la suite de ce chapitre.

6.2 Échanges de sites dans des graphes à sites

Pour rester simple, le cadre général des groupes de symétries sur les graphes à sites ne sera pas décrit. Au lieu de cela, la présentation se concentre sur le cas particulier où les symétries prennent la forme d'échanges de sites. Le but de cette section est de définir l'effet de la permutation de sites dans les différents éléments du langage Kappa, c'est à dire les motifs, les plongements, les règles et l'application des règles.

Les échanges de sites forment un groupe. Ainsi il est possible de les composer et chaque échange de sites admet un échange inverse. De plus, ce groupe est engendré par les échanges entre deux sites (qui consistent à permuter l'état de deux sites dans l'occurrence d'une protéine). Aussi, seul l'effet des échanges entre deux sites sera décrit, l'effet des autres transformations pouvant être déduit en décomposant ces échanges en séquences d'échanges élémentaires de sites.

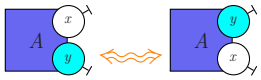
6.2.1 Échanges d'une paire de sites dans la configuration d'une espèce biochimique

Il est possible d'échanger l'état d'une paire de sites de la même occurrence d'un agent dans la configuration d'une espèce biochimique. Plutôt que d'échanger leurs états, il est possible de représenter de manière duale cette transformation en échangeant le nom des sites correspondants. Dans la représentation graphique, le site initial et le site avec lequel il est échangé gardent les mêmes positions pour représenter explicitement l'origine des sites transformés avant cet échange.

Des exemples sont donnés en figure 6.16. Deux configurations d'espèces biochimiques qui peuvent être obtenues l'une à partir de l'autre en permutant éventuellement une paire de sites dans une ou plusieurs occurrences d'une même protéine, seront appelées *symétriques*, en précisant éventuellement par quel type d'échanges de sites.

Exemple 6.2.1 *En figure 6.16, les échanges de sites sont appliqués aux configurations des espèces biochimiques de l'exemple étudiés dans ce chapitre (voir en figure 6.1). En figure 6.16(a), est montré l'effet des échanges de sites sur les occurrences du monomère de la protéine A. Il n'y a donc qu'un échange possible, puisqu'il n'y a qu'une occurrence de la protéine A. Cet échange est dessiné en orange. Les deux sites de l'occurrence de la protéine A sont nécessairement libres, sinon ce ne serait pas un monomère. Du coup, l'échange des deux sites laisse la configuration de cette espèce biochimique inchangée (puisqu'en effet la position des sites dans une occurrence de protéine n'a pas de signification particulière en Kappa).*

Pour les occurrences de dimères de la protéine A, il est possible d'échanger les sites de l'occurrence de gauche de la protéine A, les sites de l'occurrence de droite A, ou les deux. En figure 6.16(b), est montré, en vert, l'effet de l'échange des sites de l'occurrence gauche de la protéine A. En figure 6.16(c), est montré, en rouge, l'effet de



(a) Échange des sites dans les occurrences du monomère de la protéine A.

(b) Échange des sites de l'occurrence gauche de la protéine A dans les occurrences de dimères de la protéine A.

(c) Échange des sites de l'occurrence droite de la protéine A dans les occurrences de dimères de la protéine A.

(d) Échanges des sites des deux occurrences de la protéine A dans les occurrences de dimères de la protéine A.

Figure 6.16: Effet des échanges de sites sur les différentes configurations d'espèces biochimiques dans le modèle jouet.

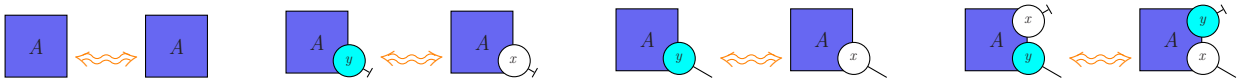
l'échange des sites de l'occurrence droite de la protéine A. Enfin en figure 6.16(d), est montré, en marron, l'effet de la combinaison des deux échanges de sites. L'ordre n'a pas d'influence sur le résultat. L'échange des sites de deux occurrences de la protéine A dans une occurrence du dimère asymétrique donne également une occurrence de dimère asymétrique, comme le montre la partie droite de la figure 6.16(d). Hormis ce cas particulier, l'application des échanges de sites aux autres configurations de dimères les transforme. Essentiellement, les liaisons entre deux sites identiques deviennent des liaisons entre deux sites différents lorsque les sites sont échangés dans une seule occurrence de la protéine A (voir en figure 6.16(b) et en figure 6.16(c)), alors que réciproquement, les liaisons entre deux sites différents deviennent des liaisons entre deux sites identiques (le site en question dépend de quel côté l'échange est appliqué). Lorsque les échanges de sites sont appliqués simultanément aux deux occurrences de la protéine A (voir en figure 6.16(d)), l'effet est de transformer un dimère formé par une liaison entre les deux sites x des occurrences de la protéine A en un dimère formé par une liaison entre les deux sites y des occurrences de la protéine A, et réciproquement.

6.2.2 Échanges d'une paire de sites dans les occurrences d'une protéine d'un motif

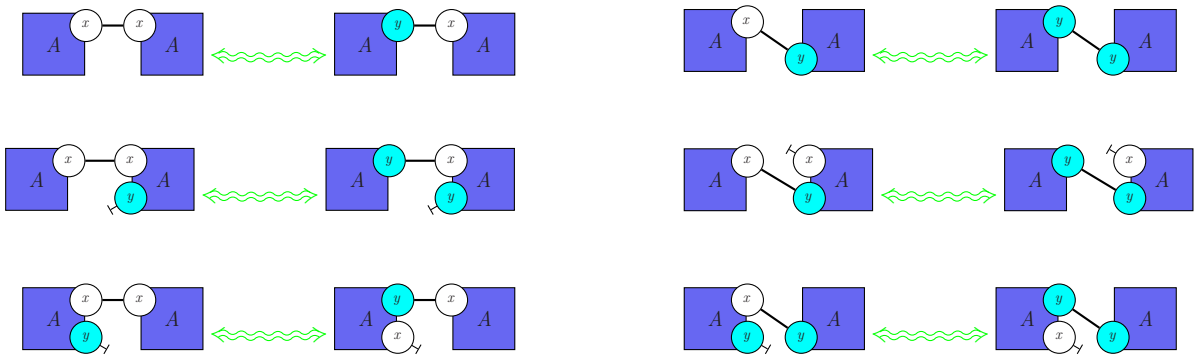
Il est possible d'échanger une paire de sites dans une occurrence de protéine dans un motif. Pour cela il suffit de choisir une occurrence de protéine dans un motif, et deux sites dans l'interface de la sorte de la protéine correspondante. Si l'occurrence de protéine ne contient aucun de ces deux sites, alors le motif reste tel qu'il est. Si l'occurrence de protéine contient exactement un des deux sites, alors celui-ci est remplacé par l'autre site en gardant ses éventuels états d'activation et de liaison. Enfin, si l'occurrence de protéine contient les deux sites, alors l'un est remplacé par l'autre, et réciproquement. De ce fait, les deux sites échangent leurs éventuels états d'activation et de liaison. Le motif reste inchangé si les deux sites avaient les mêmes états. Deux motifs qui peuvent être obtenus l'un à partir de l'autre en échangeant éventuellement des paires de sites dans une ou plusieurs occurrences d'une même protéine sont appelés *motifs symétriques*. On dit également que l'un est le symétrique de l'autre, en précisant éventuellement par quel type d'échanges de sites.

Exemple 6.2.2 *L'effet des échanges de sites dans les occurrences des protéines des motifs connexes qui ne forment pas des configurations d'espèces biochimiques (parce l'une de leur occurrence de la protéine A y est partiellement définie) est décrit en figure 6.17.*

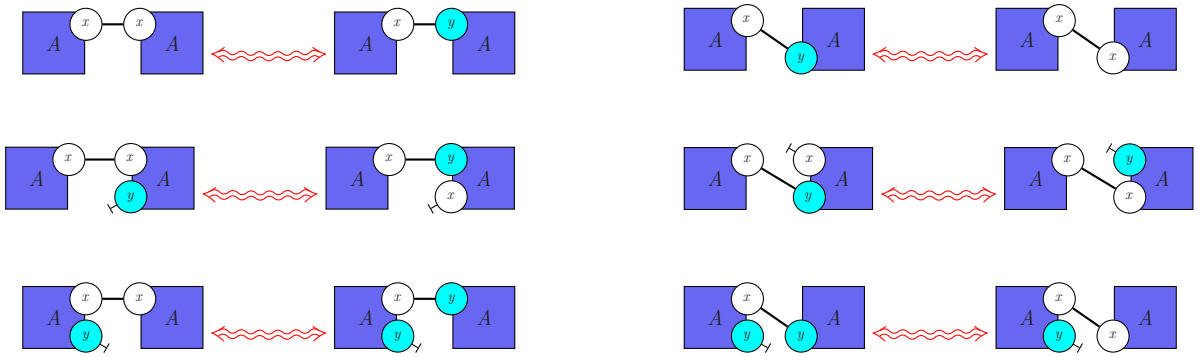
L'effet des échanges de sites dans les motifs formés d'une seule occurrence de la protéine A est décrit en figure 6.17(a). Les transformations correspondantes sont représentées par des doubles flèches ondulées oranges. Permuter les sites x et y dans l'occurrence de la protéine A dans un motif qui ne contient aucun site, laisse cette occurrence inchangée. Lorsqu'une occurrence de la protéine A ne contient que le site x, son symétrique ne contient que le site y en préservant son état de liaison (qui peut être libre ou lié sans préciser à quel site)



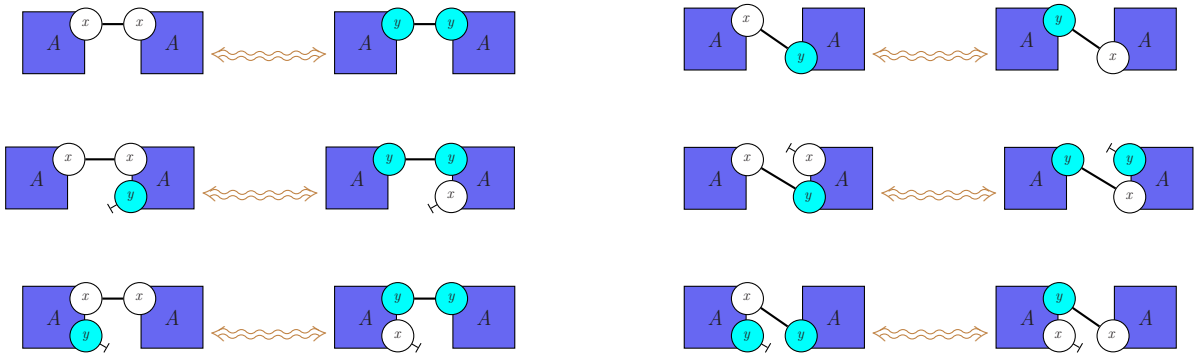
(a) Échange des sites sur les autres motifs avec une seule occurrence de la protéine A partiellement définie.



(b) Échange des sites de l'occurrence gauche de la protéine A dans les autres motifs formés de deux occurrences de la protéine A partiellement définies.



(c) Échange des sites de l'occurrence droite de la protéine A dans les autres motifs formés de deux occurrences de la protéine A partiellement définies.



(d) Échanges des sites des deux occurrences de la protéine A dans les autres motifs formés de deux occurrences de la protéine A partiellement définies.

Figure 6.17: Effet des échanges de sites dans les motifs connexes qui ne sont pas des configurations d'espèces biochimiques.

et réciproquement. Le cas d'une instance du monomère de la protéine A dont les deux sites sont libres, a déjà été vu. En effet, c'est la configuration d'une espèce biochimique (voir en figure 6.16(a)). Par ailleurs, il est impossible dans ce modèle de trouver une instance de la protéine A avec ses deux sites liés, il reste donc uniquement le cas d'une instance de protéine avec deux sites, l'un libre, et l'autre lié à un site inconnu. Dans ce cas, échanger les sites revient à rendre libre le site qui était lié et lié celui qui était libre.

En figure 6.17(b), sont représentés par des doubles flèches ondulées vertes, l'échange des sites x et y dans l'occurrence de gauche de la protéine A dans des motifs formés de deux occurrences de la protéine A connectées entre elles. Les liens symétriques sont transformés en des liens asymétriques, et réciproquement (le type de lien étant défini par la site qui était lié dans l'occurrence de droite de la protéine A). De plus, si le motif contenait un site libre, celui-ci est conservé mais change de nom s'il était situé dans l'occurrence de gauche de la protéine.

En figure 6.17(c), sont représentés par des doubles flèches ondulées rouges, l'échange des sites x et y dans l'occurrence de droite de la protéine A dans des motifs formés de deux occurrences de la protéine A connectées entre elles. Les liens symétriques sont transformés en des liens asymétriques, et réciproquement (le type de lien étant défini par la site qui était lié dans l'occurrence de droite de la protéine A). De plus, si le motif contenait un site libre, celui-ci est conservé mais change de nom s'il était situé dans l'occurrence de gauche de la protéine.

En figure 6.17(d), sont représentés par des doubles flèches marrons, les échanges des sites x et y dans les deux occurrences de la protéine A dans les motifs formés de deux occurrences de la protéine A connectées entre elles. Les liens symétriques restent symétriques mais sont portés par l'autre site. Les liens asymétriques restent asymétriques (mais le rôle des deux occurrences de la protéine A sont échangés). De plus, si le motif contenait un site libre, celui-ci est conservé mais change de nom tout en restant dans la même occurrence de la protéine A.

6.2.3 Échanges d'une paire de sites dans les occurrences d'une protéine dans une règle

Il est maintenant possible de définir l'effet des échanges de sites dans une règle. Il faut pour cela utiliser la relation implicite entre les occurrences de protéines du membre gauche et les occurrences de protéines du membre droit de la règle. Dans le sous-ensemble du langage Kappa considéré, comme les créations et les dégradations d'occurrences de protéines ne sont pas permises, chaque occurrence de protéine dans le membre gauche d'une règle correspond à une unique occurrence de protéine dans le membre droit, et réciproquement.

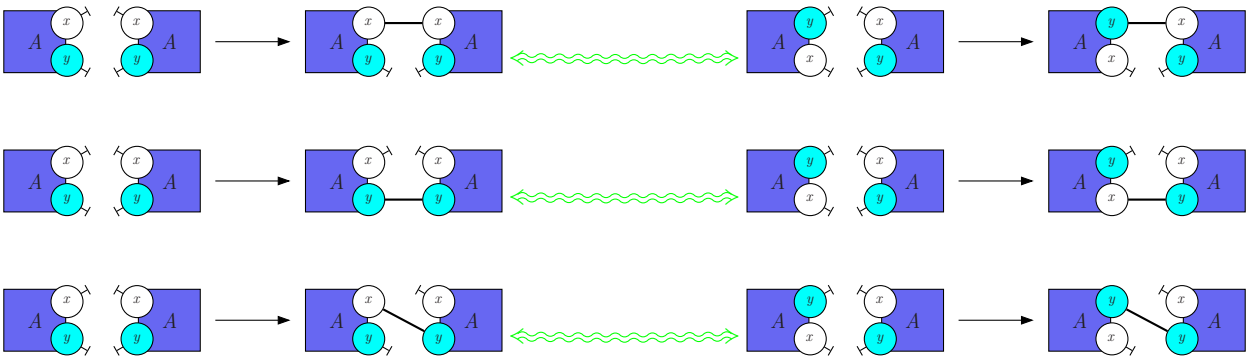
De ce fait, pour échanger les sites dans une règle, il suffit d'appliquer les mêmes échanges de sites au membre gauche et au membre droit de cette règle, c'est à dire appliquer, pour chaque occurrence de la protéine en question, le même échange de sites à gauche et à droite. Le résultat d'une telle transformation reste une règle d'interaction valide. Celle-ci est appelée le *symétrique* de la règle initiale, en précisant éventuellement par quel type d'échanges de sites.

Exemple 6.2.3 *Dans le modèle étudié dans ce chapitre, chaque règle, qu'elle soit d'association ou de dissociation, fait intervenir deux occurrences de la protéine A et chacune de ces occurrences documentent l'état de liaison des sites x et y. Il y a donc quatre échanges de sites possibles à opérer sur chaque règle. Leur effet est décrit en figure 6.18 et en figure 6.19.*

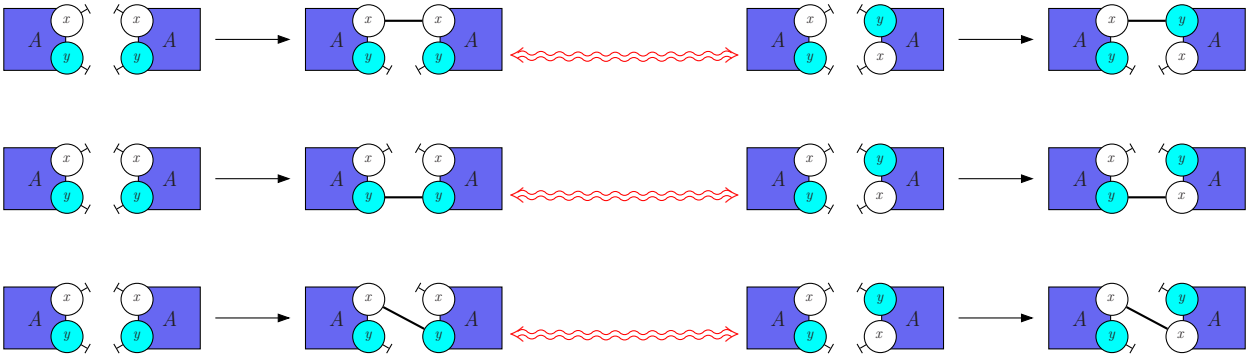
Ne rien échanger laisse les règles telles quelles. Il est possible d'échanger l'état des sites x et y dans la première occurrence de la protéine A, ce qui est représenté en figure 6.18(a) (pour les règles de liaison) et en figure 6.19(a) (pour les règles de dissociation) par des doubles flèches ondulées de couleur verte. Il est possible d'échanger l'état des sites x et y dans la seconde occurrence de la protéine A, ce qui est représenté en figure 6.18(b) (pour les règles de liaison) et en figure 6.19(b) (pour les règles de dissociation) par des doubles flèches ondulées de couleur rouge. Enfin, il est possible de permuter l'état des sites x et y simultanément dans les deux occurrences de la protéine A(a) *seconde occurrence de la protéine A, ce qui est représenté en figure 6.18(c) (pour les règles de liaison) et en figure 6.19(c) (pour les règles de dissociation) par des doubles flèches ondulées de couleur marron.*

6.2.4 Échanges de sites dans les occurrences d'une protéine dans un plongement

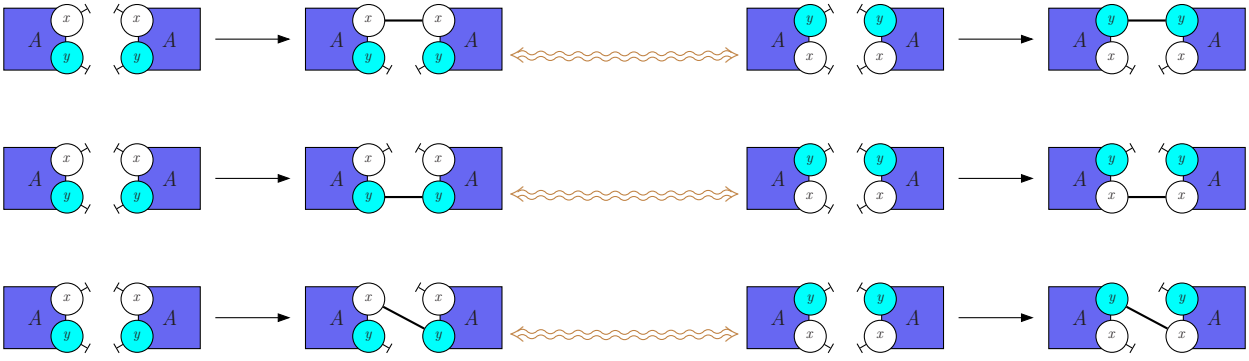
L'application d'une règle repose sur l'existence d'un plongement du membre gauche de cette règle et le graphe représentant soit l'état du système, soit des configurations d'espèces biochimiques à transformer. Pour raisonner sur les échanges de sites au cours de l'exécution d'un modèle, nous avons donc besoin de définir le symétrique d'un plongement d'un motif dans un autre par un échange de sites. Les échanges de sites dans un plongement



(a) Échange des sites de l'occurrence gauche de la protéine A dans les règles de liaison.

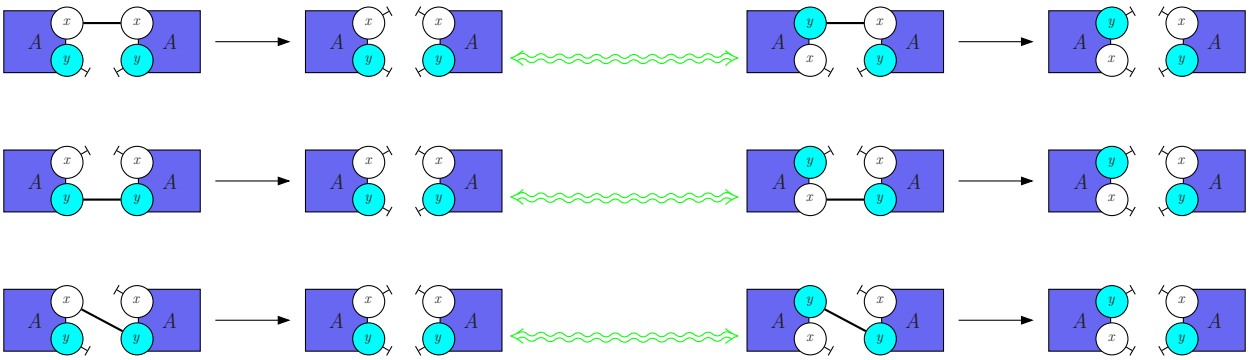


(b) Échange des sites de l'occurrence droite de la protéine A dans les règles de liaison.

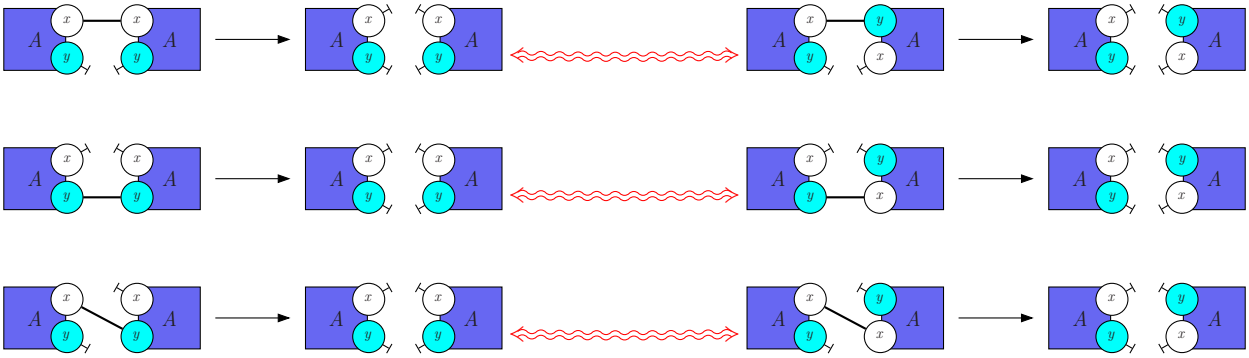


(c) Échanges des sites des deux occurrences de la protéine A dans les règles de liaison.

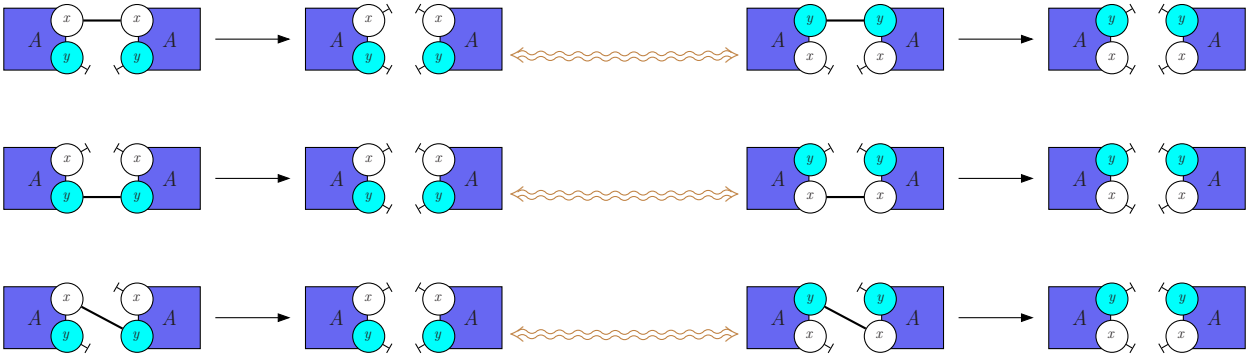
Figure 6.18: Effet des échanges de sites dans les règles de liaison du modèle jouet.



(a) Échange des sites de l'occurrence gauche de la protéine A dans les règles de dissociation.

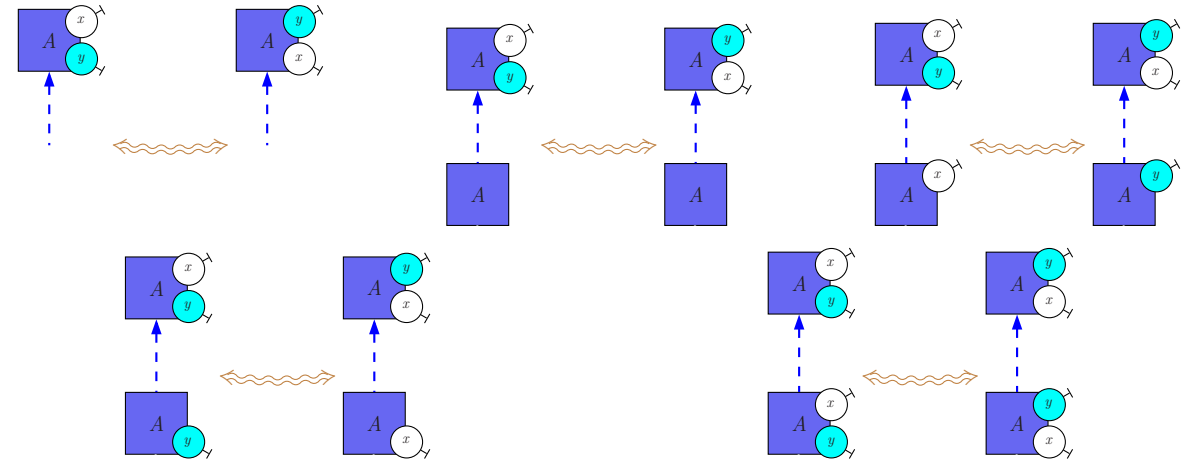


(b) Échange des sites de l'occurrence droite de la protéine A dans les règles de liaison.

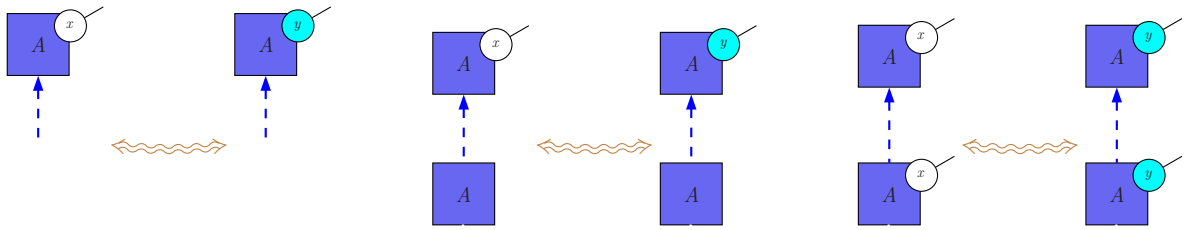


(c) Échanges des sites des deux occurrences de la protéine A dans les règles de dissociation.

Figure 6.19: Effet des échanges de sites dans les règles de dissociation du modèle jouet.



(a) Échange des sites dans les plongements vers les occurrences du monomère de la protéine A.



(b) Échange des sites dans les plongements vers le motif constitué d'une occurrence de la protéine A avec le site x lié et le site y non documenté.

Figure 6.20: Effet des échanges de sites sur les plongements vers des motifs constitués d'une seule occurrence de la protéine A.

d'un motif source dans un motif cible, est entièrement caractérisé par les échanges de sites appliqués sur le motif cible. Ensuite, chaque occurrence de protéines dans le motif source se voit appliquer le même échange de sites que ceux qui ont été appliqués à son image dans le motif cible. Ainsi, les échanges de sites qui s'appliquent aux occurrences de protéines du motif image sans antécédent n'ont pas d'effet sur le motif source.

Par construction, la correspondance entre les occurrences de protéines du motif source et ceux du motif cible est toujours valide dans leur symétrique. Du coup, celle-ci induit également un plongement entre le symétrique du motif source et le symétrique du motif cible. Nous appelons ce plongement le *symétrique* du plongement initial par échange des sites en question. On dit alors que ces deux plongements sont symétriques pour le type d'échanges de sites en question.

Exemple 6.2.4 *Des exemples de plongements symétriques sont donnés en figure 6.20, 6.21, 6.22, et en figure 6.23.*

En figure 6.20, est décrit l'effet des échanges de sites sur des plongements vers les occurrences du monomère de la protéine A. L'image de ces plongements est donc un graphe constitué d'une occurrence de la protéine A, dont les deux sites sont libres. Quant à la source de ces plongements, cinq graphes à sites sont considérés. Ce sont en fait ceux qui se plongent dans le graphe cible, si l'on s'interdit de représenter un site d'interaction sans état de liaison. De ce fait, le graphe source peut être vide ou comporter une unique occurrence de la protéine A. Dans le second cas, l'occurrence de la protéine peut comporter aucun site d'interaction, l'un des deux sites d'interaction ou les deux sites d'interaction. Par ailleurs, les éventuels sites d'interaction sont libres car, d'une part, seuls les graphes dont chaque site documente son état de liaison sont considérés dans cet exemple et que, d'autre part, les sites ne peuvent être que libres pour que le motif source se plonge dans le graphe cible.

Quel est alors l'effet d'échanger les sites sur le graphe image du plongement ? Cet échange ne peut avoir une action visible que si des sites sont présents dans le graphe source. Ainsi, lorsque la source du plongement est le graphe vide ou lorsque cette source est constitué d'une occurrence de la protéine A sans site documenté, alors

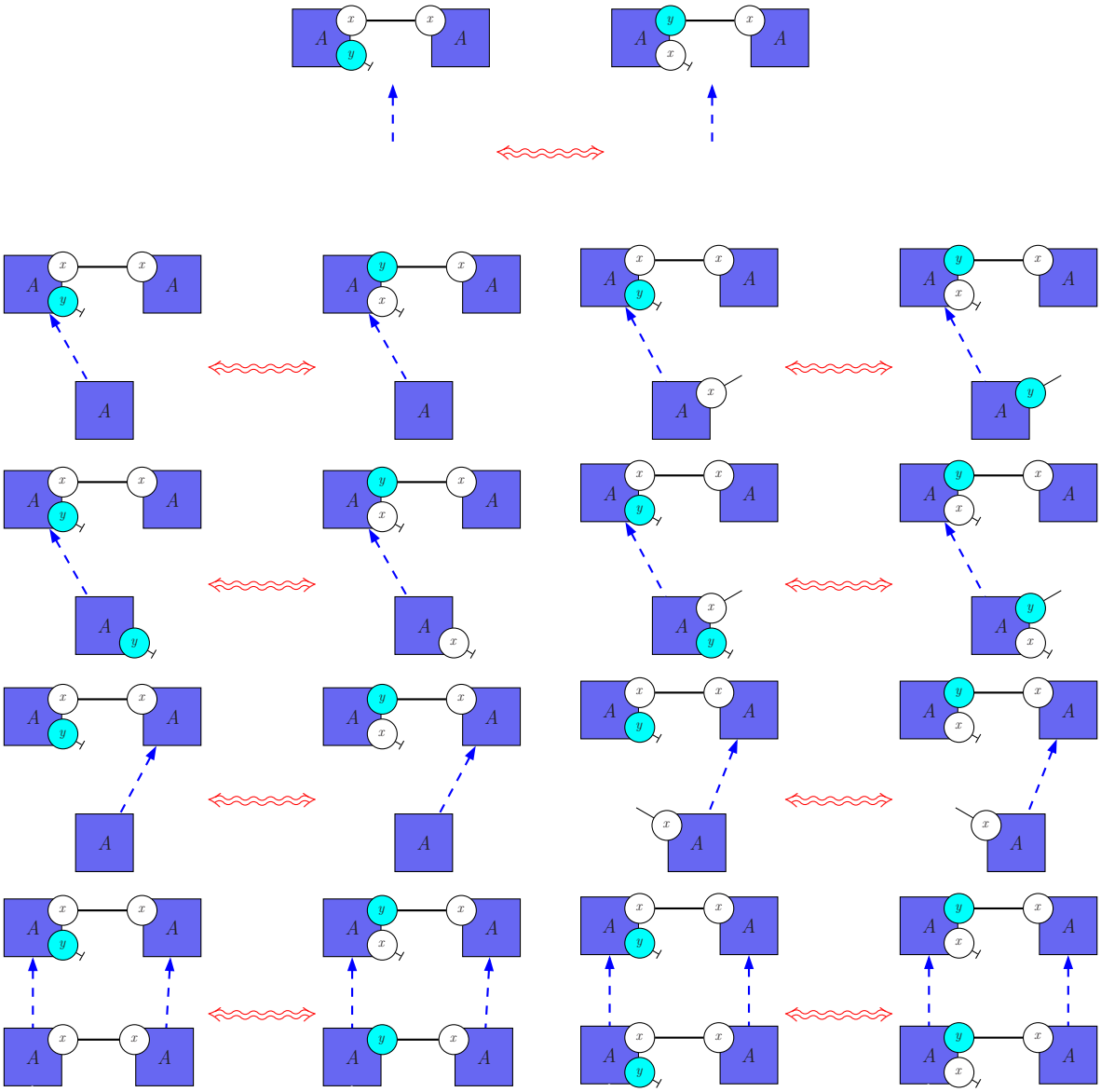


Figure 6.21: Effet de l'échange des sites de l'occurrence gauche de la protéine A dans un plongement vers un motifs formé de deux occurrences de la protéine A.

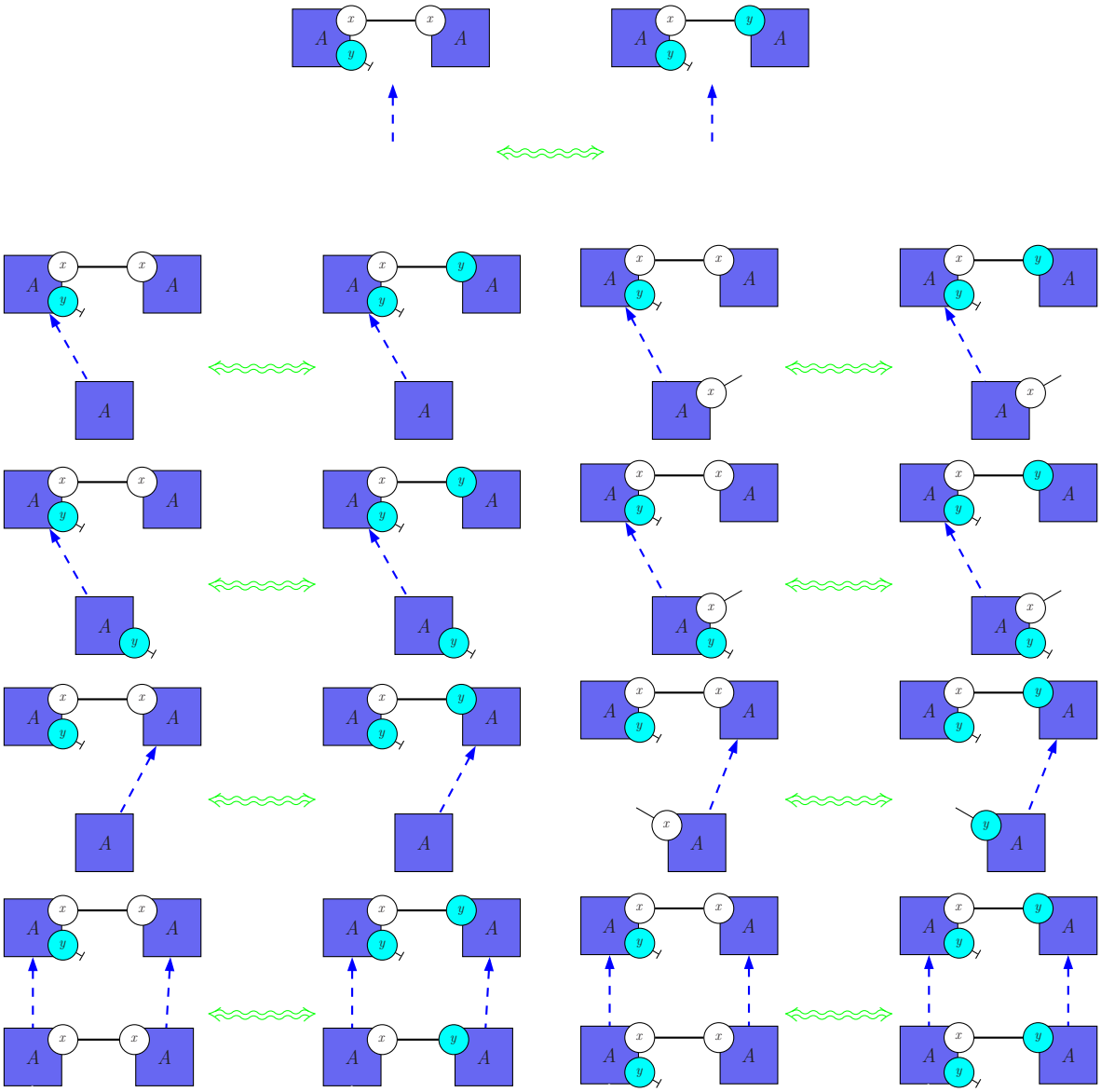


Figure 6.22: Effet de l'échange des sites de l'occurrence droite de la protéine A dans un plongement vers un motifs formé de deux occurrences de la protéine A.

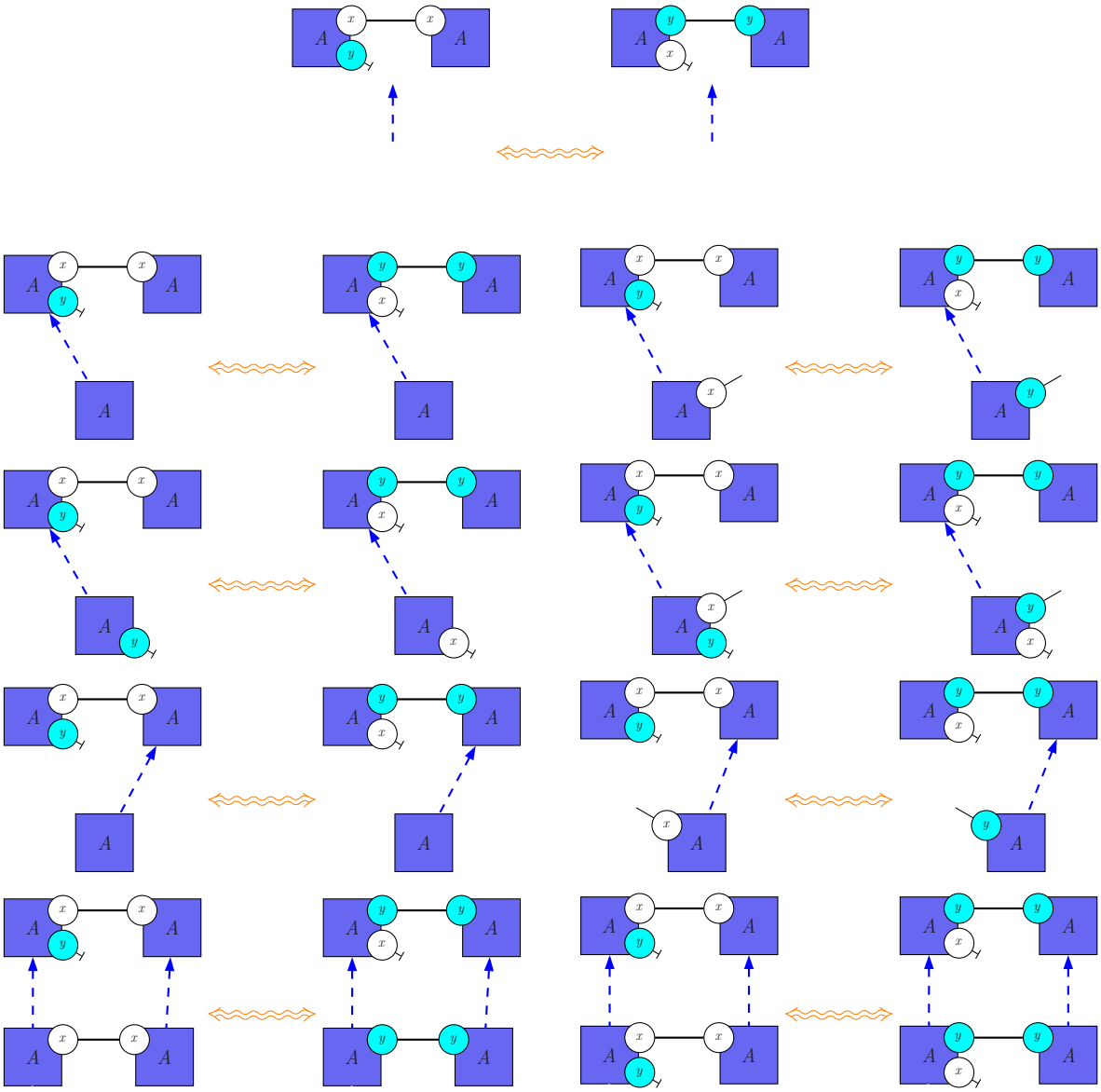


Figure 6.23: Effet des échanges des sites de chacune des occurrences de la protéine A dans un plongement vers un motifs formé de deux occurrences de la protéine A.

l'échange de site est sans effet. Dans les autres cas, comme le rôle des sites x et y est échangé, les occurrences du site x deviennent des occurrences du site y, et réciproquement. Dans notre cas, les deux sites sont libres dans le graphe cible. De ce fait, si le graphe source ne contient que le site x, l'échange de sites donne un graphe qui ne contient que le site y et si le graphe source ne contient que le site y, l'échange de sites donne un graphe qui ne contient que le site x. Enfin, lorsque le graphe source contient les deux sites, il reste inchangé.

En figure 6.20(b), est représenté l'effet de l'échange des sites sur des plongements vers un motif constitué d'une seule occurrence de la protéine A, dont le site x est lié et dont le site y n'est pas documenté. Toujours en ne considérant que des graphes dont l'état des sites est documenté, il reste trois graphes qui se plongent dans ce graphe cible : le graphe vide ; le graphe constitué d'une occurrence de la protéine A, sans site, et le graphe constitué d'une occurrence de la protéine A avec le site x lié (le graphe cible). Ici encore, dans les deux premiers cas, le graphe source n'est pas affecté par l'échange des sites, puisque ce graphe n'a pas de sites. Dans le troisième cas, le site reste lié, mais devient un site y.

En figure 6.21, 6.22 et en figure 6.23, est représenté l'effet des échanges de sites dans des plongements vers un motif formé de deux occurrences de la protéine A connectées par leurs sites x respectifs, dont le site y d'une occurrence est libre et dont le site y de l'autre occurrence n'est pas documenté. Comme il y a deux occurrences de l'agent A, il est possible d'échanger les sites x et y dans la première instance de la protéine A, dans la seconde ou dans les deux. L'échange des sites x et y dans la première occurrence de la protéine A est représenté par des doubles flèches ondulées rouges en figure 6.21. L'échange des sites x et y dans la seconde occurrence de la protéine A est représenté par des doubles flèches ondulées vertes en figure 6.22. Les échanges simultanés des sites x et y dans les deux occurrences de la protéine A est représenté par des doubles flèches ondulées oranges en figure 6.23. Dans chacune de ces trois figures, les plongements sont représentées en commençant par le cas où la source est le graphe vide, puis par les cas où la source est un graphe formé d'une seule occurrence de la protéine A se plongeant dans la première occurrence de la protéine A du graphe cible (ce qui représente quatre cas, selon que l'état du site x et l'état du site y sont représentés ou non dans ce motif). Ensuite sont considérés les cas où la source est un graphe formé d'une seule occurrence de la protéine A se plongeant dans la seconde occurrence de la protéine A du graphe cible (ce qui représente deux cas, selon que l'état du site x est documenté ou non dans ce motif ; l'état du site y ne peut pas être documenté puisque le motif doit se plonger dans la seconde occurrence de la protéine A dans le motif cible). Enfin, sont considérés les cas où le motif source contient deux occurrences de la protéine A. Pour limiter le nombre de cas, seuls sont considérés ceux où les deux occurrences sont liées entre elles par leur site x respectifs. De ce fait, cela restreint l'ensemble de ces cas à deux, selon que l'état du site y de la première occurrence de la protéine A est représenté ou non.

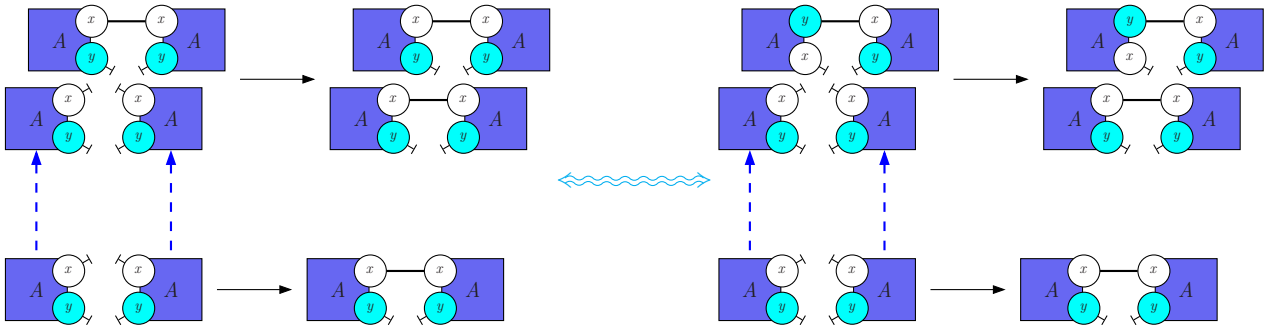
Voici maintenant les remarques principales qui peuvent être faites sur ces exemples. D'une part, le motif source n'est transformé que s'il contient au moins une occurrence de la protéine A avec au moins un site et qui se plonge dans une occurrence de la protéine A du motif cible dont les sites x et y sont échangés. De ce fait, quand le motif source est vide, il n'est pas transformé. De plus, quand le motif source ne contient qu'une occurrence de la protéine A qui se plonge dans la seconde occurrence de la protéine A dans la motif cible et que seuls les sites x et y de la première occurrence de la protéine A sont échangés, le motif source n'est pas modifié non plus (voir en figure 6.21). C'est aussi le cas lorsque le motif source ne contient qu'une occurrence de la protéine A et que celle-ci se plonge dans la première occurrence de la protéine A alors que seuls les sites x et y de la seconde occurrence de la protéine A sont échangés (voir en figure 6.22).

6.2.5 Échanges d'une paire de sites dans les occurrences d'une protéine dans le raffinement d'une règle

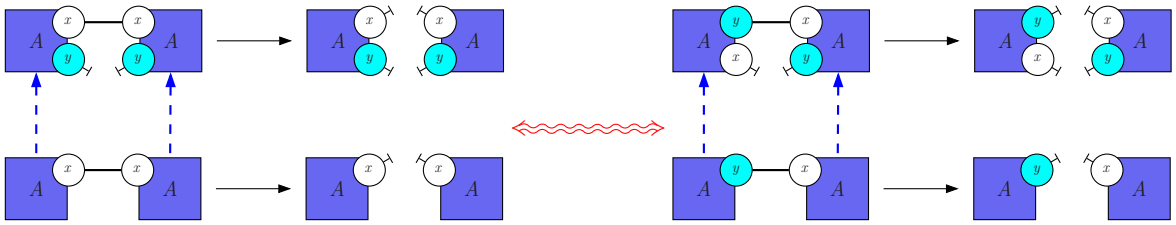
Les pas de calculs de la sémantique stochastique et les réactions de la sémantique différentielle sont obtenus en raffinant les règles (voir en section 5.1.3). Le but de cette partie est de définir l'effet de l'échange d'une paire de sites dans le raffinement d'une règle.

Appliquer des échanges de sites sur le raffinement d'une règle consiste à appliquer ces échanges de sites sur la règle raffinée, puis à propager ces échanges le long du plongement entre le membre gauche de la règle initiale et le membre gauche de la règle raffinée. Il suffit alors d'appliquer à la règle initiale l'effet de ces échanges de sites. Le symétrique de la règle raffinée par ces échanges de sites est alors un raffinement du symétrique de la règle initiale le long du symétrique du plongement entre le membre gauche de la règle initiale et le membre gauche de sa règle raffinée. Ce raffinement est alors appelé le *symétrique* du raffinement initiale par les échanges de la paire de sites en question.

Exemple 6.2.5 *En Figure 6.24 sont donnés deux exemples de raffinement de pas de calculs.*



(a) Échange d'une paire de sites dans l'application d'une règle d'association.



(b) Échange d'une paire de sites dans l'application d'une règle de dissociation.

Figure 6.24: Deux exemples de raffinements de pas de calculs. Dans le premier 6.24(a) est appliqué un échange de sites dans un pas de calcul consistant à associer les deux sites x de deux occurrences du monomère de la protéine A dans un état formé de quatre occurrences de la protéine A, deux libres et deux liées par leurs sites x respectifs. L'échange de sites porte sur l'occurrence gauche de la protéine A dans l'occurrence du dimère. De ce fait, son action ne concerne pas la règle d'interaction qui est donc gardée telle quelle lorsque cet échange est restreint à celle-ci. Dans le second 6.24(b), un échange de sites est appliqué à une réaction - ou un pas de calcul - qui consiste à enlever une liaison entre les deux sites x de deux occurrences de la protéine A. L'échange porte sur l'occurrence de gauche de la protéine A. Le résultat de cet échange est donc une dissociation entre deux occurrences de la protéine A liées par le site y de l'occurrence gauche de la protéine et leur site x de l'occurrence droite de la protéine. Ceci impose de remplacer la règle d'interaction qui a engendré la réaction - ou le pas de calcul - par sa règle symétrique.

Le premier exemple illustre le cas où les échanges de sites ont lieu dans des occurrences de protéines qui ne concernent par l'application de la règle d'interaction. Ainsi, en Figure 6.24(a) est considéré l'application d'une règle d'association symétrique entre les sites x de deux occurrences de la protéine A, à un état formé de quatre occurrences de la protéine A, deux libres (qui vont donc se lier par l'application de la règle) et deux déjà liées par leurs sites x respectifs. L'échange de sites opère dans l'occurrence gauche des deux occurrences de la protéine A déjà liées. Ces deux occurrences ayant été ajoutées pour le raffinement de la règle d'interaction, ils ne sont pas l'image d'occurrences de la protéine A du membre gauche de cette règle. De ce fait, le pas de calcul initial et son symétrique sont tous deux engendrés par la même règle d'interaction.

Dans le second exemple, donné en Figure 6.24(b), un échange entre une paire de sites intervient dans un pas de calcul qui consiste à retirer la liaison entre les deux sites x de deux occurrences de la protéine A. Cela peut être pris non seulement comme un pas de calcul, mais aussi comme une réaction, car aucune nouvelle composante connexe n'a été ajoutée dans le raffinement de la règle. L'échange de sites porte sur l'occurrence gauche de l'occurrence de la protéine A. Le résultat est un pas de calcul - ou une réaction - qui retire une liaison entre le site x d'une occurrence de la protéine A et le site y d'une autre occurence de cette protéine. Pour ce faire, il faut utiliser une règle de déliaison asymétrique. C'est bien ce qui est obtenu en restreignant l'échange de sites le long du plongement entre le membre gauche de la règle d'interaction et le membre gauche du pas de calcul - ou de la réaction.

Dans le symétrique d'un raffinement de règle, le symétrique de la règle de base n'est pas nécessairement une règle du modèle en question. Ainsi la réaction ou le pas de calcul obtenu n'est pas nécessairement une réaction ou un pas de calcul de ce modèle, et même dans ce cas, des conditions sur les constantes d'interaction correspondantes doivent être vérifiées pour engendrer des propriétés quantitatives sur le comportement du

modèle. Ceci sera l'objet de la section 6.3.1.2.

6.3 Symétries et conséquences sur le comportement des modèles

Après avoir examiné l'effet des échanges de sites sur les motifs, leurs plongements et les règles d'interaction, il est temps de se définir sous quels conditions un ensemble de règles, un état de la sémantique différentielle ou une distribution d'états de la sémantique stochastique sont symétriques par rapport à un type d'échanges de sites dans une ou plusieurs occurrences d'une protéine. Ceci sera fait en examinant leurs *orbites* qui sont obtenues à partir d'un objet de départ en appliquant toutes les combinaisons d'échanges de sites possibles. Les symétries permettent de déduire des propriétés sur le comportement des modèles, ce qui prendra la forme de bisimulations en avant ou en arrière.

6.3.1 Ensembles de règles

Sous certaines conditions, un ensemble de règle peut être symétrique par rapport à un type d'échanges de sites. Ceci permet de considérer les différentes sémantiques à échange de sites près.

6.3.1.1 Orbites de règles d'interaction

L'*orbite d'une règle d'interaction*, selon une paire de sites, regroupe toutes les manières de transformer cette règle, en échangeant cette paire de sites dans un sous-ensemble des occurrences de la protéine en question. Au sein d'une même orbite, la règle de départ n'est pas importante. En effet, les échanges de sites éventuels entre deux règles peuvent s'annuler en échangeant les deux sites de nouveau dans les mêmes occurrences de la protéine A. De ce fait, les orbites regroupent les règles en classes d'équivalence. Elle définissent les règles qui établissent la même transformation sur les graphes à sites, modulo les échanges de la paire de sites en question.

Exemple 6.3.1 *Les règles du modèle dessinées en section 6.1.1.1 présentent exactement deux orbites. Celles-ci sont données en figure 6.25. La première orbite, voir en figure 6.25(a), regroupe les règles qui établissent une liaison entre les deux occurrences de la protéine A, alors que la seconde, voir en figure 6.25(b), regroupe les règles qui brisent une liaison.*

À noter que chaque règle de liaison et de dissociation symétrique apparaît exactement une fois dans son orbite respective. Les règles asymétriques apparaissent exactement deux fois, quitte à échanger l'occurrence de droite et de gauche dans ces règles.

6.3.1.2 Ensembles symétriques de règles

Pour vérifier si deux sites jouent exactement le même rôle dans un ensemble de règles, il suffit d'en inspecter les orbites.

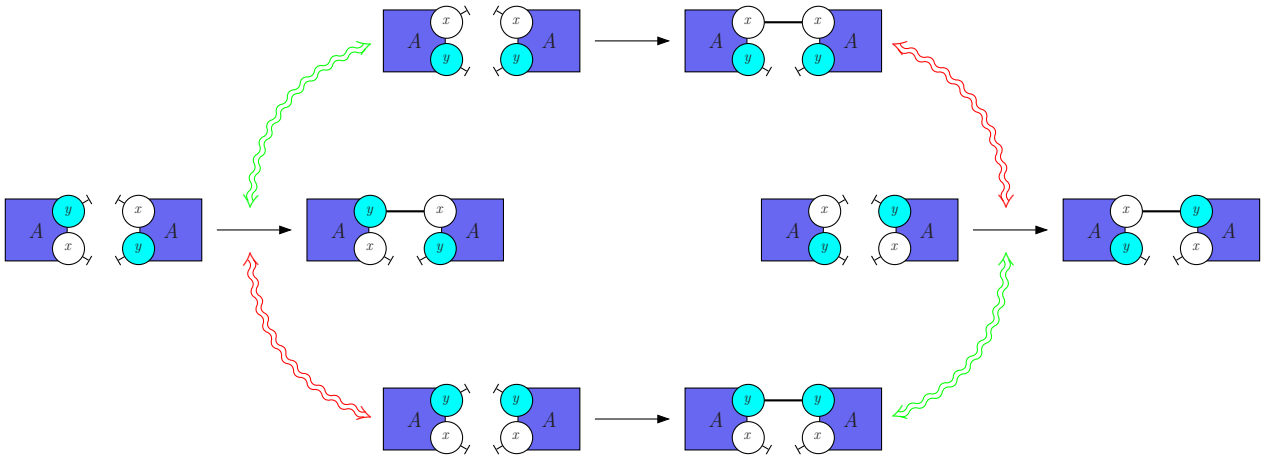
Dans un premier temps, il faut vérifier que si un modèle comporte une règle donnée, alors elle comporte également toutes les autres règles de son orbite, c'est à dire, toutes les règles qui peuvent être obtenues en échangeant ces deux sites dans une ou plusieurs occurrences de la protéine en question. Toutefois, une règle peut compter pour une règle équivalente par isomorphisme de règle (voir page 21). Dans un second temps, il faut s'assurer que leurs constantes d'interaction sont proportionnelles à leurs nombres d'apparitions dans l'orbite.

Exemple 6.3.2 *Dans l'exemple de la section 6.1.1.1, les règles symétriques (celles qui agissent sur deux sites x ou sur deux sites y) apparaissent chacune une fois dans leurs orbites respectives, alors que les règles asymétriques (celles qui agissent sur un site x et un site y) apparaissent sous deux formes isomorphes quitte à changer l'ordre des occurrences de la protéine A. Ainsi pour que l'ensemble des règles du modèle soit symétrique, il faut que les contraintes suivantes soient vérifiées :*

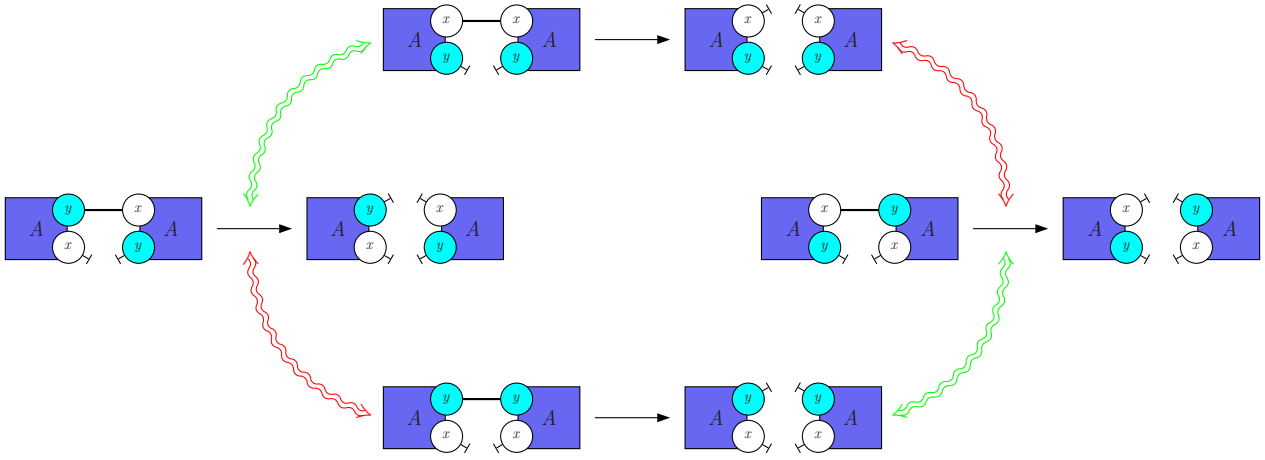
$$\frac{k_{xx}}{1} = \frac{k_{yy}}{1} = \frac{k_{xy}}{2} \qquad \textit{et} \qquad \frac{k^d_{xx}}{1} = \frac{k^d_{yy}}{1} = \frac{k^d_{xy}}{2}.$$

Dans ces fractions, le dénominateur représente le nombre de formes de la règle (à permutation des agents près) qui apparaissent dans les catégories de règles.

De ce fait, les contantes d'interaction pour les liaisons symétriques doivent être égales, alors que la constante d'interaction pour les liaisons asymétriques doit être égale au double de cette valeur commune. Il en



(a) Les règles d'association sous forme d'orbite.



(b) Les règles de dissociation sous forme d'orbite.

Figure 6.25: Quitte à échanger les sites x et y dans certaines occurrences de la protéine A, les règles se regroupent en deux catégories. Les règles d'association sont représentées en figure 6.25(a), alors que les règles de dissociation sont représentées en figure 6.25(b). L'effet des échanges de sites est représenté par des doubles flèches ondulées. Les échanges des sites x et y dans l'occurrence gauche de la protéine sont représentés en vert, alors ceux dans l'occurrence de droite sont représentés en rouge.

est de même pour les constantes d'interaction des règles de dissociation. On retrouve ainsi les contraintes sur les constantes d'association et les constantes de dissociation qui avaient permis, d'une part, de quotienter la sémantique différentielle et la sémantique stochastique en section 6.1.3.1 et, d'autre part, de faire émerger des invariants quantitatifs en section 6.1.3.2.

6.3.1.3 Orbites de motifs

Les ensembles symétriques de règles assurent des propriétés sur les pas de calculs. Toutefois, de telles règles ne produisent pas les différents états dans les mêmes proportions. Il convient donc d'inspecter les *orbites* formées par les graphes à sites lorsque des paires de sites sont échangées pour comprendre les rapports de proportionnalités qui vont apparaître. Ces orbites sont, une fois encore, obtenues en échangeant une paire de sites dans une occurrence d'une protéine donnée dans un motif.

Exemple 6.3.3 *Dans le cadre de l'exemple du modèle introduit en section 6.1.1.1, l'exemple d'une orbite pour un motif est donné en figure 6.26. Ce motif en question est formé de trois occurrences de la protéine A. Une occurrence, celle du dessous, est libre, alors que les deux autres, au dessus, sont liées. Comme il y a trois*

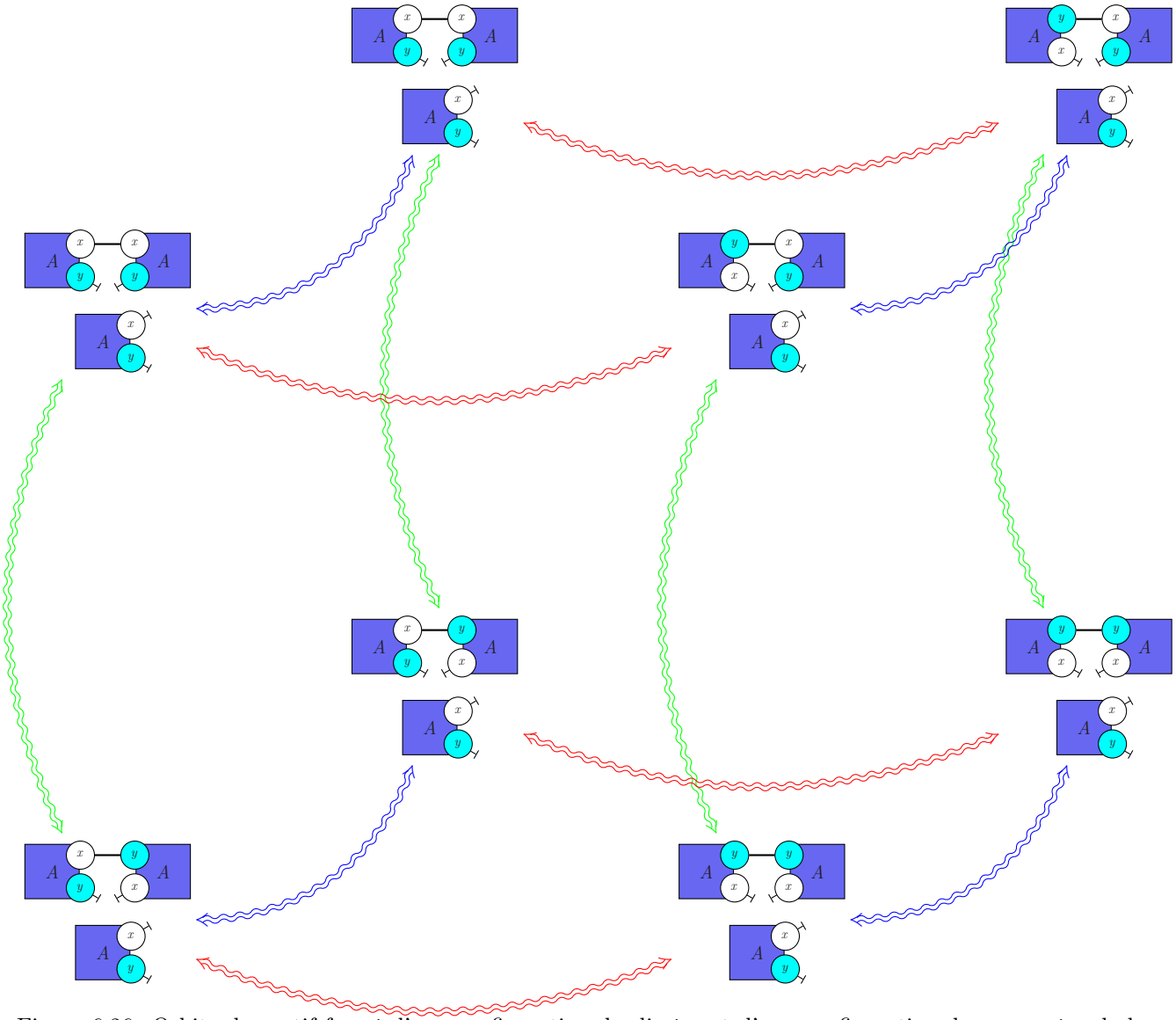
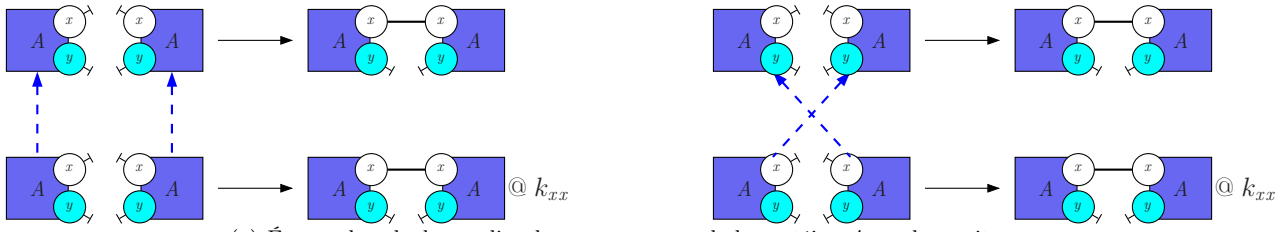
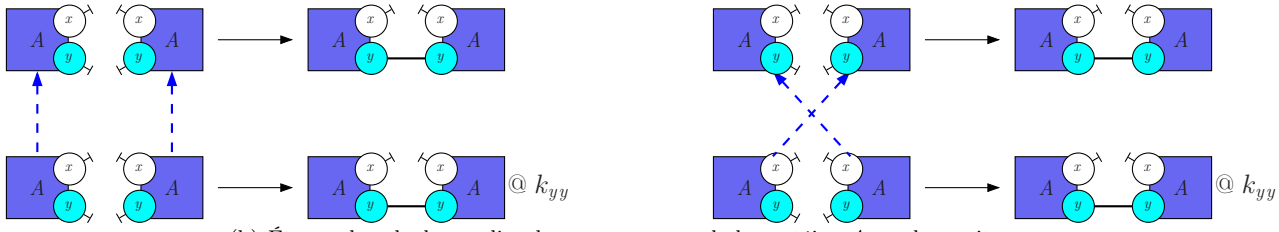


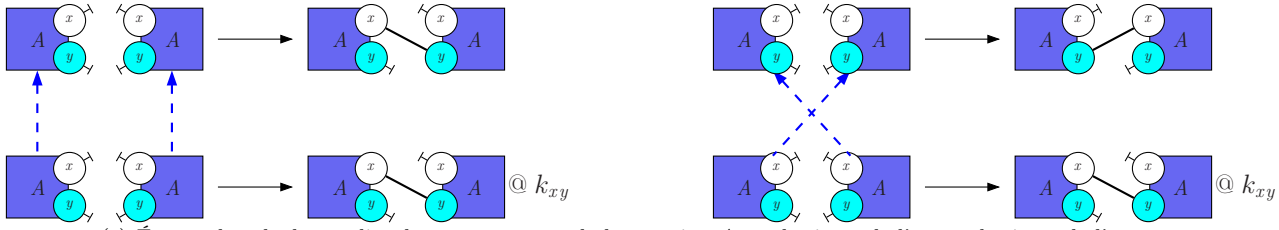
Figure 6.26: Orbite du motif formé d'une configuration du dimère et d'une configuration du monomère de la protéine A.



(a) Étapes de calcul pour lier deux occurrences de la protéine A par leurs sites x.



(b) Étapes de calcul pour lier deux occurrences de la protéine A par leurs sites y.



(c) Étapes de calcul pour lier deux occurrences de la protéine A par le site x de l'une et le site y de l'autre.

Figure 6.27: Les différentes étapes de calcul pour lier deux occurrences libres de la protéine A.

occurrences de la protéine A dans lesquelles les sites x et y peuvent être échangés, l'orbite de ce graphe à sites peut être décrite sous la forme d'un cube, chacune des dimensions correspondant à l'échange des sites dans l'une de ces trois occurrences. Les échanges des sites x et y dans l'occurrence de gauche de la protéine A dans les dimères sont représentés en rouge, ceux dans l'occurrence de droite en vert. Enfin, les échanges des sites x et y dans l'occurrence du monomère de la protéine A sont représentés en bleu. Dans ce dernier cas, comme les deux sites des occurrences de la protéine A sont libres, leur échange ne change pas le motif.

Au sein de l'orbite, le graphe à sites formé de deux occurrences de la protéine A liées entre-elles par leurs sites x et d'une occurrence libre, apparaît 2 fois. Il en est de même pour le graphe à sites formé de deux occurrences de la protéine A liées entre-elles par leurs sites y et d'une occurrence libre. Enfin, le graphe à sites formés de deux occurrences de la protéine A liées entre-elles par un site x et un site y et d'une occurrence libre apparaît, lui, 4 fois.

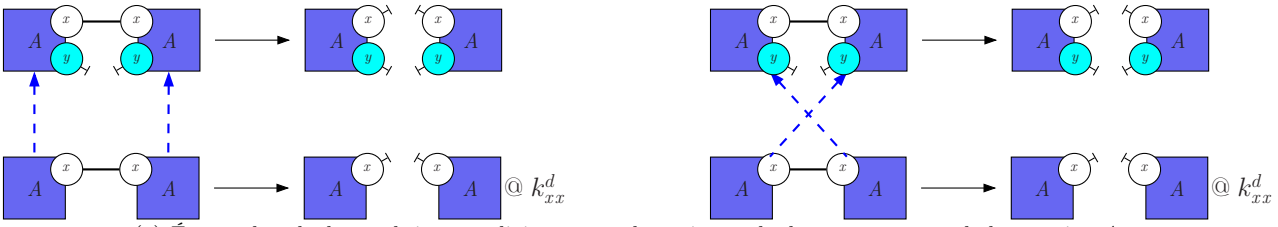
6.3.1.4 Propriété fondamentale

La propriété fondamentale des ensembles symétriques de règles peut désormais être formulée.

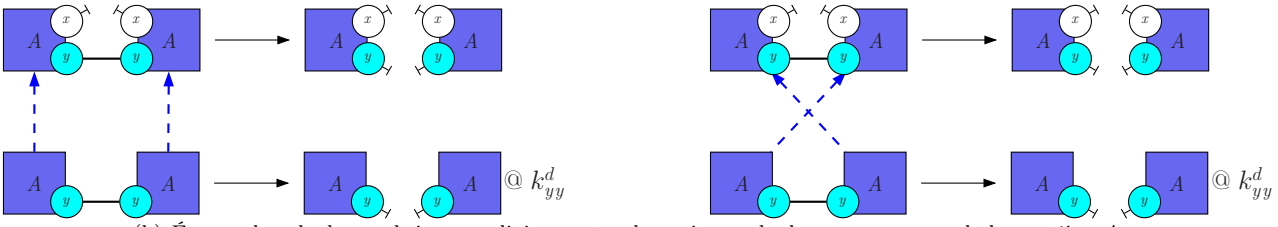
Théorème 6.3.1 (Bisimulation avant-arrière) *Dans un modèle dont l'ensemble des règles est symétrique par rapport à un type d'échanges de sites dans les occurrences d'une protéine donnée, étant donné un pas de calcul entre un état q et un état q', et le symétrique de ce pas de calcul entre l'état q_σ et un état q'_σ, alors les quantités $\tau(\{q\},\{q'\})$ et $\tau(\{q_\sigma\},\{q'_\sigma\})$ définies respectivement comme la somme pour chaque pas de calcul entre l'état q (resp. q_σ) et un état équivalent à q' (resp. q'_σ) à isomorphisme près (voir page 18) de la constante d'interaction de la règle qui a engendré ce pas de calcul sont proportionnelles au nombre d'apparitions $w(q)$ et $w(q')$ d'états isomorphes à q et à q' dans leurs orbites respectives :*

$$w(q') \cdot \tau(\{q\},\{q'\}) = w(q) \cdot \tau(\{q_\sigma\},\{q'_\sigma\}).$$

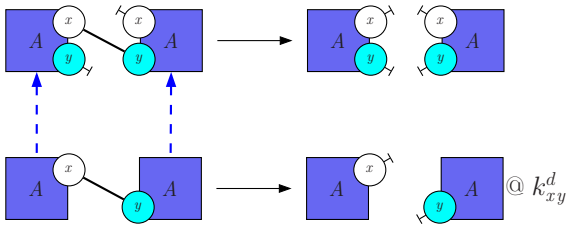
Exemple 6.3.4 *Une illustration du théorème 6.3.1 est donnée en examinant la somme des constantes d'interaction*



(a) Étapes de calcul pour briser une liaison entre deux sites x de deux occurrences de la protéine A.



(b) Étapes de calcul pour briser une liaison entre deux sites y de deux occurrences de la protéine A.



(c) Étape de calcul pour briser une liaison asymétrique entre deux occurrences de la protéine A.

Figure 6.28: Les différentes étapes de calcul pour dissocier deux occurrences liées de la protéine A.

pour les différents types de liaison et les dissociation entre les occurrences de la protéine A dans le modèle introduit en section 6.1.1.1.

- *Lorsque l'ensemble des règles est symétrique, les liaisons entre un site x et un site y se forment deux fois plus souvent que celles entre deux sites x et que celles entre deux sites y.*

  Étant données deux occurrences libres de la protéine A, chacune des trois règles de liaison peut s'appliquer selon deux plongements (voir en figure 6.27). Ceci donne comme somme des constantes d'interaction $2 \cdot k_{xx}$ *pour une liaison entre les deux sites x,* $2 \cdot k_{yy}$ *pour une liaison entre les deux sites y, et* $2 \cdot k_{xy}$ *pour une liaison entre un site x et un site y. Or dans les ensembles symétriques de règles, les constantes* k_{xx} *et* k_{yy} *sont égales, alors que la constante* k_{xy} *est égale au double de la constante* k_{xx}*. Les sommes des constantes d'interaction sont donc bien proportionnelles au nombre d'occurrences des configurations du dimère de la protéine A dans leur orbite, sachant que la configuration asymétrique y apparaît deux fois, alors que chaque configuration symétrique y apparaît exactement une fois.*

- *Lorsque l'ensemble des règles est symétrique, les liaisons entre les occurrences de la protéine A se séparent toutes au même rythme.*

  En effet, les occurrences des configurations symétrique des dimères se séparent avec la constante d'interaction k^d_{xx} *ou* k^d_{yy} *selon leur type. Cependant, il existe deux manières d'appliquer la règle correspondante, ce qui donne une somme de constantes égale à* $2 \cdot k^d_{xx}$ *ou* $2 \cdot k^d_{yy}$*. Par ailleurs, il n'existe qu'une manière d'appliquer la règle de dissociation pour les configurations de dimère asymétriques, ce qui donne une somme égale à* k^d_{xy}*. Comme dans les ensembles symétriques de règles, les deux constantes* k^d_{xx} *et* k^d_{yy} *sont égales et que la constante* k^d_{xy} *est égale au double de la constante* k^d_{xx}*, le comportement attendu est bien obtenu.*

Une conséquence directe du théorème 6.3.1, est qu'un ensemble symétrique de règles engendre à la fois une bisimulation en avant et une bisimulation en arrière.

Ainsi la propriété suivante s'obtient en sommant pour les états cibles, sur tous les états d'une classe de symétrie.

Propriété 6.3.1 (Bisimulation en avant) *Dans un modèle dont l'ensemble des règles est symétrique par rapport à un type d'échanges de sites dans les occurrences d'une certaine protéine, étant donnés deux états symétriques q et q', et un autre état q'' d'équivalence d'états C, alors la somme $\tau(\{q\}, [q''])$ et la somme $\tau'(\{q'\}, [q''])$ définies respectivement comme la somme pour chaque pas de calcul entre l'état q (resp. q') et un état isomorphe à un état symétrique à q'' de la constante d'interaction de la règle d'interaction qui a engendré ce pas de calcul sont égales :*

$$\tau(\{q\}, [q'']) = \tau'(\{q'\}, [q'']).$$

La propriété 6.3.1 indique que deux états symétriques ont toujours le même comportement vis à vis d'une classe d'équivalence d'états.

La propriété suivante s'obtient, elle, en sommant pour les états départs, sur tous les états d'une classe de symétrie.

Propriété 6.3.2 (Bisimulation en arrière) *Dans un modèle dont l'ensemble des règles est symétrique par rapport à un type d'échanges de sites dans les occurrences d'une certaine protéine, étant donnés deux états symétriques q et q', alors :*

1. *d'une part, la somme $\tau(\{q\}, \_)$ et la somme $\tau(\{q'\}, \_)$ définies respectivement comme la somme pour chaque pas de calcul issu de l'état q (resp. q') des constantes d'interaction de la règle qui a engendré ce pas de calcul sont égales (quelque soit l'état d'arrivée), sont égales :*

$$\tau(\{q\}, \_) = \tau'(\{q'\}, \_);$$

2. *et, d'autre part, pour tout état q'', la somme $\tau([q''], \{q\})$ et la somme $\tau'([q''], \{q'\})$ définies respectivement comme la somme pour chaque pas de calcul entre un état symétrique à l'état q'' et l'état q (resp. q') à isomorphisme près, de la constante d'interaction de la règle qui a engendré ce pas de calcul, sont proportionnelles au nombre d'apparition $w(q)$ et $w(q')$ d'états isomorphes à q et à q' :*

$$w(q') \cdot \tau([q''], \{q\}) = w(q) \cdot \tau'([q''], \{q'\}).$$

La propriété 6.3.2 indique que d'une part, le système avance à la même vitesse pour toute paire d'états symétriques, et que d'autre part, le comportement du système préserve les rapports de proportionnalité.

6.3.2 Effet des symétries dans la sémantique différentielle

Le but de cette partie est de donner sous quelles conditions suffisantes des symétries dans un ensemble de règles d'interaction induisent des propriétés quantitatives dans le modèle différentiel induit par ces règles.

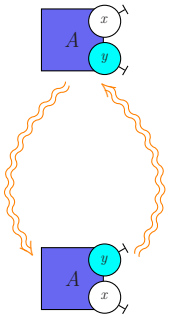
6.3.2.1 Orbites des configurations d'espèces biochimiques

Pour ce faire, il faut regarder l'effet des échanges de sites sur les configurations d'espèces biochimiques. En effet, ce sont leurs quantités respectives qui constituent les variables des systèmes différentiels engendrés par les ensembles de règles de réécriture de graphes à sites.

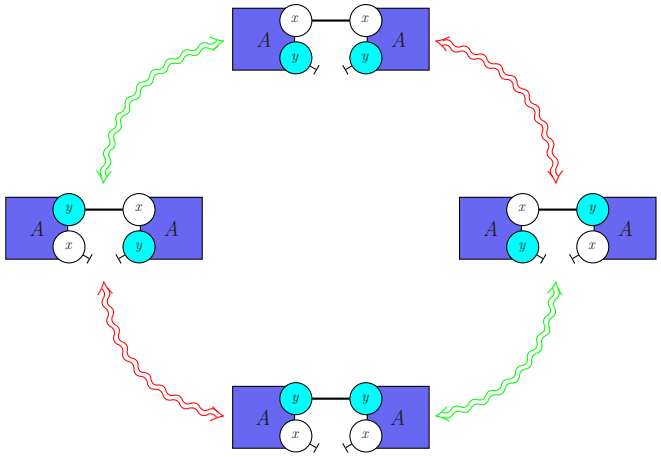
Les configurations d'espèces biochimiques sont des motifs particuliers. Comme le symétrique d'une configuration d'une espèce biochimique par échange de sites est lui-même la configuration d'une espèce biochimique, il est possible de les regrouper en classe d'équivalence en calculant leurs *orbites*.

Exemple 6.3.5 *Dans le modèle donné en section 6.1.1.1, les configurations d'espèces biochimiques se classent exactement en deux classes d'équivalence, lorsque l'état de liaison des sites x et y des occurrences de la protéine A est échangé : l'unique configuration du monomère, d'une part, et les configurations du dimère, d'autre part. Ces deux orbites sont dessinées en Figure 6.29.*

Les monomères existent sous une seule forme (voir en figure 6.29(a)). Ils ne sont pas modifiés par l'échange de leurs sites x et y. Les dimères existent sous trois formes, selon que la liaison entre les deux occurrences de la protéine A se fasse sur les deux sites x des occurrences de la protéine, sur leurs deux sites y ou sur un site x et un site y (voir en Figure 6.29(b)). Cette dernière forme est représentée deux fois, sous deux formes isomorphes quitte à intervertir l'ordre des deux occurrences de la protéine A.



(a) Orbite de l'unique configuration du monomère de la protéine A.



(b) Orbite des configurations du dimère de la protéine A.

Figure 6.29: Quitte à permuter les sites x et y dans les occurrences de la protéine A, les configurations d'espèces biochimiques se classent en deux catégories, celles du monomère de la protéine A (voir en figure 6.29(a)) et celles du dimère de la protéine A (voir en figure 6.29(b)). Pour la configuration du monomère, la seule transformation possible consiste à permuter les sites de l'unique occurrence de la protéine (ce qui est dessiné avec une double flèche ondulée orange). Ceci ne change par la configuation de la protéine puisque l'ordre des sites n'a pas d'importance dans le langage Kappa. Pour les configurations du dimère, les rôles des sites peuvent être échangés dans l'occurrence gauche (ce qui est dessiné avec une double flèche ondulée verte) ou dans l'occurrence droite (rouge). Quitte à changer l'ordre des agents et des sites de liaisons, les deux représentations graphiques de configurations asymétriques du dimère sont isomorphes.

6.3.2.2 Bisimulation en avant

Le but de cette section est de montrer que les symétries dans un ensemble de règles d'interactions induisent une relation de bisimulation sur les états des modèles différentiels qu'elles induisent.

Cette relation consiste à regrouper la quantité des configurations symétriques d'espèces biochimiques. Il suffit pour cela de choisir une représentante pour chaque classe d'équivalence. Ceci permet de définir une fonction r qui associe à chaque configuration d'une espèce biochimique, la configuration symétrique qui est la représentante de sa classe d'équivalence. La fonction r est une fonction idempotente (ainsi $r \circ r = r$). Cette fonction permet de définir deux fonctions sur les états différentiels. La première, P_r regroupe la quantité des configurations d'espèces biochimiques sur les représentantes de leurs classes d'équivalence respectives, alors que la seconde, Z_r annule la quantité des configurations d'espèces biochimiques qui ne sont pas les représentantes de leurs classes d'équivalence respectives.

Ceci donne les définitions suivantes :

$$P_r(\rho) = \begin{cases} \mathcal{V} & \to & \mathbb{R} & \\ v & \mapsto & \sum\{\rho(v') \mid r(v') = v\} & \text{si } r(v) = v \\ v & \mapsto & 0 & \text{sinon,} \end{cases} \qquad Z_r(\rho) = \begin{cases} \mathcal{V} & \to & \mathbb{R} & \\ v & \mapsto & \rho(v) & \text{si } r(v) = v \\ v & \mapsto & 0 & \text{sinon.} \end{cases}$$

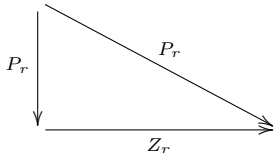
Exemple 6.3.6 *Dans le modèle jouet introduit en section 6.1.1.1, il existe deux classes d'équivalence de configurations d'espèces biochimiques pour les échanges entre les sites x et y dans les occurrences de la protéine A, selon le nombre d'occurrences de cette protéine dans l'espèce biochimique en question. La classe de configuration du monomère est réduite à un élément qui est donc la représentante de sa classe d'équivalence. Pour les configurations du dimère de la protéine A, la configuration choisie comme représentante de sa classe d'équivalence est celle où les deux occurrences de la protéine A sont liées par leurs sites x respectifs.*

Les fonctions P_r et Z_r prennent alors les valeurs suivantes :

$$P_r(\rho) = \begin{cases} A & \mapsto & \rho(A) \\ A.x - x.A & \mapsto & \rho(A.x - x.A) + \rho(A.y - y.A) + \rho(A.x - y.A) \\ A.y - y.A & \mapsto & 0 \\ A.x - y.A & \mapsto & 0 \end{cases}$$

$$Z_r(\rho) = \begin{cases} A & \mapsto & \rho(A) \\ A.x - x.A & \mapsto & \rho(A.x - x.A) \\ A.y - y.A & \mapsto & 0 \\ A.x - y.A & \mapsto & 0 \end{cases}$$

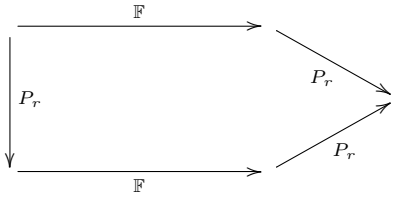
Les fonctions P_r et Z_r sont des projections linéaires. En effet, elles sont toutes deux linéaires et idempotentes. Par ailleurs, elles ont la même image, à savoir l'ensemble des états qui associent la quantité 0 à toutes les configurations d'espèces biochimiques qui ne sont pas les représentantes de leurs classes d'équivalence respectives. De ce fait, le diagramme suivant :



commute, ce qui signifie que les fonctions P_r et $Z_r \circ P_r$ sont égales.

Lorsque l'ensemble de règles est symétriques, la fonction P_r induit une bisimulation sur l'ensemble des états de la sémantique différentielle du modèle engendrée par ces règles. En effet, pour deux états X et X' tels que $P_r(X) = P_r(X')$, le théorème 6.3.1 permet de montrer que l'égalité suivante : $P_r(\mathbb{F}(X)) = P_r(\mathbb{F}(X'))$ est également satisfaite.

De manière équivalente, le diagramme suivant :

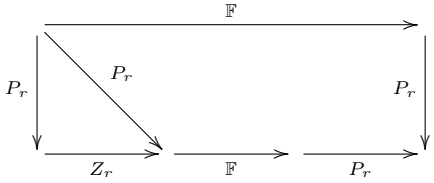


commute, ce qui signifie que les fonctions suivantes $P_r \circ \mathbb{F}$ et $P_r \circ \mathbb{F} \circ P_r$ sont égales.

Ainsi:

$$P_r \circ \mathbb{F} = P_r \circ \mathbb{F} \circ P_r$$
$$P_r \circ \mathbb{F} = P_r \circ \mathbb{F} \circ [Z_r \circ P_r]$$
$$P_r \circ \mathbb{F} = [P_r \circ \mathbb{F} \circ Z_r] \circ P_r$$

ce qui peut se résumer par la diagramme commutatif suivant :



Ce diagramme définit une réduction de modèle (voir en section 4.3) avec pour fonction d'abstraction P_r et pour contre-partie abstraite à la fonction $\mathbb{F}$ la fonction $P_r \circ \mathbb{F} \circ Z_r$.

Exemple 6.3.7 *Dans le modèle jouet introduit en section 6.1.1.1, la fonction $\mathbb{F}$ est définie de la manière suivante :*

$$\mathbb{F}(\rho) = \begin{cases} A & \mapsto 2k^d_{xy}\rho(A.x - y.A) + 2k^d_{yy}2\rho(A.y - y.A) + 2k^d_{xx}2\rho(A.x - x.A) - (2k_{yy} + 2k_{xx} + 2k_{xy})\,\rho(A)^2 \\ A.x - x.A & \mapsto 2k_{xx}\rho(A)^2 - k^d_{xx}2\rho(A.x - x.A) \\ A.y - y.A & \mapsto 2k_{yy}\rho(A)^2 - k^d_{yy}2\rho(A.y - y.A) \\ A.x - y.A & \mapsto 2k_{xy}\rho(A)^2 - k^d_{xy}\rho(A.x - y.A). \end{cases}$$

La composition à droite avec la fonction Z_r annule la contribution des configurations d'espèces biochimiques qui ne sont pas les représentantes de leurs classes d'équivalence. Cela élimine donc les occurrences des variables $A.y - y.A$ et $A.x - y.A$ dans la fonction $\mathbb{F}$.

Ainsi:

$$[\mathbb{F} \circ Z_r](\rho) = \begin{cases} A & \mapsto 2k^d_{xx}2\rho(A.x - x.A) - (2k_{yy} + 2k_{xx} + 2k_{xy})\,\rho(A)^2 \\ A.x - x.A & \mapsto k_{xx}\rho(A)^2 - k^d_{xx}2\rho(A.x - x.A) \\ A.y - y.A & \mapsto k_{yy}\rho(A)^2 \\ A.x - y.A & \mapsto k_{xy}\rho(A)^2. \end{cases}$$

Enfin, la composition à gauche avec la fonction P_r regroupe les différentes contributions sur les représentantes de leurs classes d'équivalence.

Ainsi:

$$[P_r \circ \mathbb{F} \circ Z_r](\rho) = \begin{cases} A & \mapsto 2k^d_{xx}2\,[A.x - x.A] - (2k_{yy} + 2k_{xx} + 2k_{xy})\,[A]^2 \\ A.x - x.A & \mapsto (k_{xx} + k_{yy} + k_{xy})\,[A]^2 - k^d_{xx}2\,[A.x - x.A] \\ A.y - y.A & \mapsto 0 \\ A.x - y.A & \mapsto 0. \end{cases}$$

Ce qui correspond à l'équation différentielle pour le modèle simplifié suivante :

$$\begin{cases} A_1 & \mapsto 2K^d2\rho(A_2) - 2K\rho(A_1)^2 \\ A_2 & \mapsto K\rho(A_1)^2 - 2K^d\rho(A_2). \end{cases}$$

La bisimulation en avant prouve alors les résultats observés en figure 6.14(b) et 6.14(f), à savoir que les trajectoires de concentrations dans le modèle réduit sont bien la somme des trajectoires des quantités de configurations d'espèces biochimiques correspondantes dans le modèle initial dès lors que l'ensemble de règles est symétrique et sans imposer que l'état initial le soit.

6.3.2.3 Bisimulation en arrière

Le but de cette section est de montrer que les symétries dans un ensemble de règles d'interactions permettent d'assurer des invariants sur les quantités de configurations d'espèces biochimiques, puis d'utiliser ces invariants pour éliminer des variables dans le système d'équations différentielles engendré par l'ensemble de règles en question.

Il faut dans un premier temps définir la notion de symétrie pour les états de la sémantique différentielle par rapport à un type d'échanges de sites. Les états de la sémantique différentielle sont des fonctions qui associent à chaque configuration d'espèces biochimiques, un réel positif qui représente sa quantité. Un tel état est dit symétrique par rapport à un type d'échanges de sites, si, pour toute paire de configurations symétriques d'espèces biochimiques pour ce type d'échanges de sites, les quantités présentes de ces deux configurations sont proportionnelles à leurs nombres de répétitions dans l'orbite correspondante.

Exemple 6.3.8 *L'orbite des configurations potentielles du monomère de la protéine A du modèle introduit en section 6.1.1.1 n'est formée que d'un élément. Par contre, celle pour les configurations du dimère de la protéine A comporte les configurations symétriques du dimère avec une liaison entre les deux sites x ; les configuration symétriques du dimère avec une liaison entre les deux sites y et les configuration asymétriques du dimère avec une liaison entre le site x d'une occurrence de la protéine A et le site y de l'autre occurrence de la protéine A. À isomorphisme près, cette dernière configuration apparaît deux fois dans son orbite.*

En conséquence, les états symétriques de la sémantique différentielle sont ceux tels que la quantité de configuration asymétriques des dimères représente la moitié de la quantité totale en dimère et la quantité de chacune des deux configurations symétriques du dimère de la protéine A représentent chacune un quart de la quantité totale en dimère. Il s'agit des mêmes proportions que pour le jeu de "pile" ou "face".

La propriété 6.3.2 assure, par sommation, que dans un modèle engendré par un ensemble symétrique de règles par rapport à un type d'échanges de sites, si X est un état symétrique par rapport ce type d'échanges de sites, alors il en est de même pour $\mathbb{F}(X)$. Puis, comme les combinaisons linéaires de deux états symétriques par rapport à un type d'échanges de sites, sont également symétriques par rapport à ce type d'échanges de sites, il en résulte que le système reste dans un état symétrique par rapport à ce type d'échanges de sites tout au long de son exécution.

Exemple 6.3.9 *Il est maintenant possible de justifier les constatations qui avaient été faites en figure 6.15(b). En effet, pour les paramètres cinétiques $k_{xx} = k_{yy} = 0.5$, $k_{xy} = 1$, $k^d_{xx} = k^d_{yy} = 1$ et $k^d_{xy} = 2$ et un état initial formé uniquement de la configuration du monomère de la protéine A, à la fois l'ensemble des règles et l'état initial du système sont symétriques par rapport aux échanges des sites x et y dans les occurrences de la protéine A. En conséquence, l'état du système reste symétrique pour ces échanges de sites tout au long de l'exécution du système. Ainsi la quantité de configurations du dimère avec les deux sites x reste toujours égale à la quantité de configurations du dimère avec les deux sites y, alors que la quantité de configurations du dimère formé par une liaison entre le site x d'une occurrence de la protéine et le site y de l'autre occurrence reste égale au double.*

Il est alors possible d'éliminer des variables en utilisant cette propriété. Pour cela, il faut choisir une représentante pour chaque classe d'équivalence modulo les échanges de cette paire de sites dans les configurations d'espèces biochimiques. La représentante d'une configuration d'espèce biochimique v est notée $r(v)$. Ainsi $r : \mathcal{V} \to \mathcal{V}$ est une fonction idempotente (c'est à dire $r \circ r = r$) de l'ensemble des configurations des espèces biochimiques dans lui-même. On note également $w : \mathcal{V} \to \mathbb{N} \setminus \{0\}$ la fonction qui a chaque configuration d'une espèce biochimique, associe son nombre d'occurrences dans son orbite pour les échanges de la paire de sites en question.

Deux fonctions de l'ensemble des états $\mathcal{V} \to \mathbb{R}$ vers lui-même sont définies ci-dessous. La première Eq_r a pour but de remplacer les occurrences des variables qui ne sont pas les représentantes de leurs classes d'équivalence par des occurrences de leurs représentantes respectifs avec des coefficients pour tenir des rapports de proportionnalité dans les états symétriques. La seconde Z_r a pour but d'annuler la valeur des variables qui ne sont pas les représentantes de leurs classes d'équivalence.

Ceci donne les définitions suivantes :

$$Eq_r(\rho) : \begin{cases} \mathcal{V} & \to & \mathbb{R} \\ v & \mapsto & \frac{w(v)}{w(r(v))} \cdot \rho(r(v)) \end{cases} \qquad Z_r(\rho) : \begin{cases} \mathcal{V} & \to & \mathbb{R} & \\ v & \mapsto & \rho(v) & \text{si } r(v) = v \\ v & \mapsto & 0 & \text{sinon.} \end{cases}$$

Les fonctions Eq_r et Z_r sont deux projections linéaires, mais cette fois-ci leurs images sont différentes. Grâce à la fonction Eq_r, il est possible de proposer une nouvelle manière de caractériser les états symétriques par rapport au type d'échanges de sites en question. En effet, les états ρ sont symétriques si et seulement si ils sont égaux à leurs propres images par la fonction Eq_r, c'est à dire si $Eq_r(\rho) = \rho$.

Exemple 6.3.10 *Dans le modèle jouet qui est étudié en section 6.1.1.1, il existe deux classes d'équivalence de configurations d'espèces biochimiques pour les échanges entre les sites x et y dans les occurrences de la protéine A, selon le nombre d'occurrences de cette protéine dans l'espèce biochimique en question. La classe des monomères est réduite à un élément qui est dont la représentante de sa classe d'équivalence. Pour les dimères, la configuration choisie comme représentante est celle où les deux occurrences de la protéine A sont liées par leurs sites x respectifs.*

Les fonctions Z_r et Eq_r sont définies de la manière suivante :

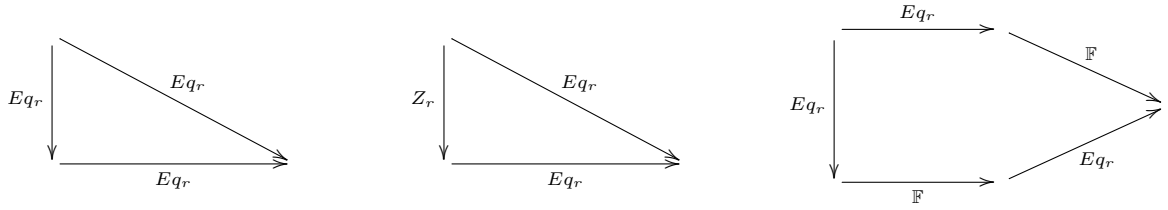
$$Eq_r(\rho) = \begin{cases} A & \mapsto & \rho(A) \\ A.x - x.A & \mapsto & \frac{1}{1} \cdot \rho(A.x - x.A) \\ A.y - y.A & \mapsto & \frac{1}{1} \cdot \rho(A.x - x.A) \\ A.x - y.A & \mapsto & \frac{2}{1} \cdot \rho(A.x - x.A) \end{cases} \qquad Z_r(\rho) = \begin{cases} A & \mapsto & \rho(A) \\ A.x - x.A & \mapsto & \rho(A.x - x.A) \\ A.y - y.A & \mapsto & 0 \\ A.x - y.A & \mapsto & 0. \end{cases}$$

Soit ρ un état, la propriété $Eq_r(\rho) = \rho$ est équivalente à l'ensemble des contraintes suivantes :

$$\begin{cases} \rho(A) &= \rho(A) \\ \rho(A.x - x.A) &= \rho(A.x - x.A) \\ \rho(A.y - y.A) &= \rho(A.x - x.A) \\ \rho(A.x - y.A) &= 2 \cdot \rho(A.x - x.A), \end{cases}$$

ce qui correspond bien aux critères pour qu'un état soit symétrique par rapport aux échanges des sites x et y dans les occurrences de la protéine A.

Les trois diagrammes suivants sont utilisés pour construire l'élimination des variables qui ne sont pas les représentantes de leurs classes d'équivalence :



Ces trois diagrammes commutent. Le premier diagramme vient du fait que la fonction Eq_r est idempotente, car c'est une projection linéaire. Ainsi $Eq_r = Eq_r \circ Eq_r$ ce qui constitue le premier diagramme. Ensuite, d'une part, le fonction Z_r ne change que la valeur des variables qui ne sont pas les représentantes de leurs classes d'équivalence et d'autre part, la fonction Eq_r ne dépend pas de ces variables. Il en découle que $Eq_r = Eq_r \circ Z_r$ ce qui constitue le second diagramme commutatif. Enfin, le troisième diagramme provient du fait que l'image par $\mathbb{F}$ d'un état symétrique est lui même symétrique. En effet, pour tout état $\rho \in \mathcal{V} \to \mathbb{R}$, l'état $Eq_r(\rho)$ est symétrique, puis comme la fonction $\mathbb{F}$ préserve l'ensemble des états symétriques, $\mathbb{F}(Eq_r(\rho))$ est un état symétrique, ce qui implique que $Eq_r(\mathbb{F}(Eq_r(\rho))) = \mathbb{F}(Eq_r(\rho))$. Cette dernière propriété est celle représentée par le troisième diagramme.

Il est alors possible de prouver l'égalité suivante :

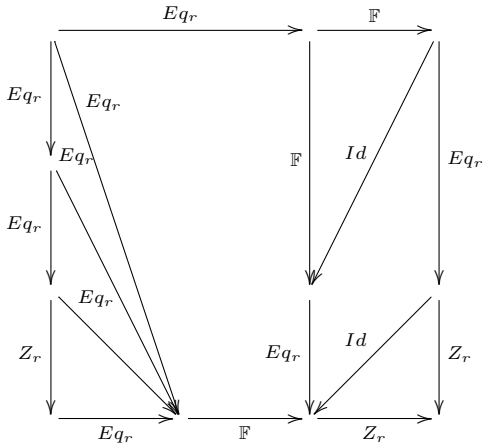
$$[Z_r \circ Eq_r] \circ [\mathbb{F}] \circ [Eq_r] = [Z_r \circ \mathbb{F} \circ Eq_r] \circ [Z_r \circ Eq_r] \circ [Eq_r]$$

est réalisée.

En effet:

$$\begin{array}{lcl} [Z_r \circ Eq_r] \circ [\mathbb{F}] \circ [Eq_r] &=& Z_r \circ [Eq_r \circ \mathbb{F} \circ Eq_r] \\ [Z_r \circ Eq_r] \circ [\mathbb{F}] \circ [Eq_r] &=& Z_r \circ [\mathbb{F} \circ Eq_r] \\ [Z_r \circ Eq_r] \circ [\mathbb{F}] \circ [Eq_r] &=& [Z_r \circ \mathbb{F} \circ [Eq_r \circ Eq_r]] \\ [Z_r \circ Eq_r] \circ [\mathbb{F}] \circ [Eq_r] &=& [Z_r \circ \mathbb{F} \circ [[Eq_r \circ Eq_r] \circ Eq_r]] \\ [Z_r \circ Eq_r] \circ [\mathbb{F}] \circ [Eq_r] &=& [Z_r \circ \mathbb{F} \circ [[[Eq_r \circ Z_r] \circ Eq_r] \circ Eq_r]] \\ [Z_r \circ Eq_r] \circ [\mathbb{F}] \circ [Eq_r] &=& [Z_r \circ \mathbb{F} \circ Eq_r] \circ [Z_r \circ Eq_r] \circ Eq_r. \end{array}$$

Ce calcul est résumé dans le diagramme commutatif suivant :



Ainsi pour tout état $\rho \in \mathcal{V} \to \mathbb{R}$ tel que $Eq_r(\rho) = \rho$, la propriété suivante :

$$[[Z_r \circ Eq_r] \circ [\mathbb{F}]]\ (\rho) = [Z_r \circ \mathbb{F} \circ Eq_r] \circ [Z_r \circ Eq_r](\rho)$$

est satisfaite.

Cette propriété est similaire à la caractérisation des réductions de modèles vues en section 4.3, avec la fonction $\mathbb{F}$ pour définir la dynamique du système différentiel initial, $Z_r \circ Eq_r$ comme fonction d'abstraction et $Z_r \circ \mathbb{F} \circ Eq_r$ comme contre-partie abstraite de la fonction $\mathbb{F}$. Il y a toutefois deux différences. D'une part, cette propriété n'est valable que pour les états symétriques (qui sont leur propre image par la fonction Eq_r). Ce n'est pas un problème puisque lorsque l'état initial est symétrique, la trajectoire reste dans l'ensemble des états symétriques. D'autre part, il peut y avoir des variables dont la contribution est nulle dans la fonction $Z_r \circ Eq_r$, puisque seules les variables qui sont les représentantes de leurs classes d'équivalence ont une contribution non nulle. Ici encore, ce n'est pas un problème, car le système simplifié préserve tout de même les asymptotes verticales. En effet, si la valeur d'une variable diverge, il en est de même pour la valeur de sa représentante de sa classe d'équivalence qui elle, apparaît avec une contribution non nulle dans le système simplifié.

De ce fait, les trajectoires maximales du systèmes simplifiées sont bien l'image point par point par la fonction d'abstraction $Z_r \circ Eq_r$ des trajectoires du système initiale qui commence dans un état symétrique.

Exemple 6.3.11 *Dans le modèle introduit en section 6.1.1.1, la fonction $\mathbb{F}$ est définie de la manière suivante :*

$$\mathbb{F}(\rho) = \begin{cases} A & \mapsto\ 2k^d_{xy}\rho(A.x - y.A) + 2k^d_{yy}2\rho(A.y - y.A) + 2k^d_{xx}2\rho(A.x - x.A) - (2k_{yy} + 2k_{xx} + 2k_{xy})\,\rho(A)^2 \\ A.x - x.A & \mapsto\ 2k_{xx}\rho(A)^2 - k^d_{xx}2\rho(A.x - x.A) \\ A.y - y.A & \mapsto\ 2k_{yy}\rho(A)^2 - k^d_{yy}2\rho(A.y - y.A) \\ A.x - y.A & \mapsto\ 2k_{xy}\rho(A)^2 - k^d_{xy}\rho(A.x - y.A). \end{cases}$$

La composition à droite avec la fonction Eq_r remplace la contribution des configurations d'espèces biochimiques qui ne sont pas les représentantes de leurs classes d'équivalence par leur représentante avec comme coefficient le rapport entre les nombres d'apparitions sous un forme isomorphe de la représentante dans l'orbite et celui de la configuration d'espèces biochimiques en question. Cela élimine donc les occurrences des variables $A.y - y.A$ et $A.x - y.A$ dans la fonction $\mathbb{F}$.

Ainsi:

$$[\mathbb{F} \circ Eq_r](\rho) = \begin{cases} A & \mapsto\ 2k^d_{xy}2\rho(A.x - x.A) + 2k^d_{yy}2\rho(A.x - x.A) + 2k^d_{xx}2\rho(A.x - x.A) \\ & \qquad\qquad - (2k_{yy} + 2k_{xx} + 2k_{xy})\,\rho(A)^2 \\ A.x - x.A & \mapsto\ 2k_{xx}\rho(A)^2 - k^d_{xx}2\rho(A.x - x.A) \\ A.y - y.A & \mapsto\ 2k_{yy}\rho(A)^2 - k^d_{yy}2\rho(A.x - x.A) \\ A.x - y.A & \mapsto\ 2k_{xy}\rho(A)^2 - k^d_{xy}2\rho(A.x - x.A) \end{cases}$$

D'où :

$$\mathbb{F} \circ Eq_r = \begin{cases} A & \mapsto\ 4(k^d_{xy} + k^d_{yy} + k^d_{xx})\rho(A.x - x.A) - (2k_{yy} + 2k_{xx} + 2k_{xy})\,\rho(A)^2 \\ A.x - x.A & \mapsto\ k_{xx}\rho(A)^2 - k^d_{xx}2\rho(A.x - x.A) \\ A.y - y.A & \mapsto\ k_{yy}\rho(A)^2 - k^d_{yy}2\rho(A.x - x.A) \\ A.x - y.A & \mapsto\ k_{xy}\rho(A)^2 - k^d_{xy}2\rho(A.x - x.A). \end{cases}$$

Enfin, la composition à gauche avec la fonction Z_r élimine les variables $A.y - y.A$ et $A.x - y.A$ qui ne sont pas les représentantes de leus classes d'équivalence.

Ainsi:

$$[Z_r \circ \mathbb{F} \circ Eq_r](\rho) = \begin{cases} A & \mapsto\ 4(k^d_{xy} + k^d_{yy} + k^d_{xx}) \cdot \rho(A.x - x.A) - (2k_{yy} + 2k_{xx} + 2k_{xy})\,\rho(A)^2 \\ A.x - x.A & \mapsto\ k_{xx}\rho(A)^2 - k^d_{xx}2\rho(A.x - x.A) \\ A.y - y.A & \mapsto\ 0 \\ A.x - y.A & \mapsto\ 0. \end{cases}$$

Ce qui correspond à l'équation différentielle pour le modèle simplifié suivante :

$$\begin{cases} A_1 & \mapsto\ 16K^d\rho(A.x - x.A) - 2K\rho(A_1)^2 \\ A.x - x.A & \mapsto\ \frac{K}{4}\rho(A_1)^2 - 2K^d\rho(A.x - x.A), \end{cases}$$

avec $K = k_{xx} + k_{xy} + k_{yy}$ *et* $K^d = k^d_{xx} = k^d_{yy} = \frac{k^d_{xy}}{2}$.

Ceci clôt la partie sur l'utilisation des symétries pour réduire la sémantique différentielle des modèles engendrés par des règles.

6.3.3 États discrets

Le but de cette partie est de dériver des propriétés similaires à celles données en Section 6.3.2, mais cette fois-ci pour la sémantique stochastique.

6.3.3.1 Bisimulation en avant

La propriété 6.3.1 implique que lorsque qu'un ensemble de règles est symétrique par rapport à un type d'échanges de sites, le système engendré par ces règles se comporte de la même manière quand un tel échange est effectué dans l'état du système. En effet, la probabilité d'aller dans une classe d'équivalence donnée est la même pour deux états symétriques.

Exemple 6.3.12 *Dans l'exemple jouet introduit en section 6.1.1.1, quitte à intervertir les sites x et y dans une ou plusieurs occurrences de la protéine A, les configurations des espèces biochimiques se répartissent en deux catégories selon qu'elles soient formées d'une ou de deux occurrences de la protéine A. Ces deux catégories sont dessinées en figure 6.29. De ce fait, les classes d'équivalence pour les échanges des sites x et y sont caractérisées par le nombre d'occurrences i de l'unique configuration du monomère de la protéine A et par la somme j des trois configurations du dimère de la protéine A.*

Cette propriété se décline sur plusieurs niveaux d'abstraction.

1. **Distribution des traces.** La sémantique de traces qui opère sur des classes d'équivalence d'états est une approximation fidèle de la sémantique de traces initiale. En effet, deux états symétriques se comportent de manière identique vis à vis des classes d'équivalence. La probabilité d'un ensemble de traces dans la sémantique quotient est alors la somme des probabilités des ensembles de traces dont ils sont l'abstraction, l'abstraction d'un ensemble de traces étant obtenue en remplaçant chaque état et chaque pas de calcul par sa classe d'équivalence et en conservant les intervalles de temps pour le délai entre chaque pas de calcul. Il faut pour cela supposer que la distribution initiale des états dans la sémantique simplifiée soit égale à la distribution obtenue en sommant la probabilité de chaque état par classe d'équivalence dans la distribution initiale des états dans la sémantique initiale.

   Exemple 6.3.13 *Ceci permet de justifier les remarques de la section 6.1.2. En effet, quite à oublier la différence entre les sites x et y dans les occurrences de la protéine A, il y a, dans le modèle de la section 6.1.1.1, uniquement deux classes d'équivalence de configurations d'espèces biochimiques, celles de l'unique configuration du monomère de la protéine A et celle qui regroupe les trois configurations du dimère de la protéine A. Ces deux classes d'équivalence peuvent se représenter comme en figure 6.8 page 101. Les règles d'interaction elles-même se représentent sous-forme de classes d'équivalence en oubliant la différence entre les sites x et y dans les occurrences de la protéine A dans leurs membres gauches et droits, ce qui donne bien les règles dessinées en figure 6.9 page 101. Les constantes de ces règles doivent assurer que la propensité d'une interaction dans un état du modèle simplifié soit la même que dans un état du modèle initial dans cette classe d'équivalence. Ainsi, la constante d'association du modèle simplifié doit être égale à la somme des trois constante d'association, alors que la règle de dissociation doit être égale à la valeur commune des deux constantes de dissociation pour les liaisons symétriques et de la moitié de la constante de dissociation pour la liaison asymétriques (le fait que le modèle soit symétrique impose que les constantes de dissociation pour les liaisons symétriques soient égales et que cette valeur commune soit la moitié de la constante de dissociation pour les liaisons asymétriques).*

2. **Simulation stochastique.** Comme dans un modèle engendré par un ensemble symétrique de règles, deux états symétriques se comportent de la même manière vis à vis des classes d'équivalence de cette symétrie, il est possible d'échanger, entre chaque étape de calcul, les paires de sites en question dans une ou plusieurs occurrences de protéines. Ceci peut avoir une incidence sur les structures de données qui sont utilisées pour représenter l'état du système en réduisant le nombre de configurations différentes d'espèces biochimiques.

Exemple 6.3.14 *Ainsi, dans le cas d'étude présenté en section 6.1.1.1, il est possible, quitte à ajuster les constantes d'interaction, d'imposer que seuls les sites x des protéines A se lient. Ceci a pour conséquence, de réduire à la fois la nombre de configurations d'espèces biochimiques accessibles et le nombre des réactions, sans changer le comportement du modèle. Ainsi, il n'y a plus que deux configurations d'espèces biochmiques, au lieu de* 4*, et* 2 *réactions au lieu de* 6*.*

3. **Équation maîtresse.** L'équation maîtresse est similaire à l'équation de la sémantique différentielle, si ce n'est qu'elle manipule des distributions de probabilité sur des états discrets, et non des quantités continues de configurations d'espèces biochimiques. Toutefois, une démarche analogue peut être utilisée pour réduire le nombre de variables dans celle-ci.

   L'équation maîtresse de système est notée $\frac{dX(t)}{dt} = A(X(t))$. Il suffit donc de choisir un représentant $r(q)$ de la classe d'équivalence de chaque état q, pour définir deux projections linéaires P_r et Z_r :

$$P_r(\rho) = \begin{cases} \mathcal{Q} & \to & \mathbb{R} & \\ q & \mapsto & \sum\{\rho(q') \mid r(q') = q\} & \text{si } r(q) = q \\ q & \mapsto & 0 & \text{sinon,} \end{cases} \qquad Z_r(q) = \begin{cases} \mathcal{Q} & \to & \mathbb{R} & \\ q & \mapsto & \rho(q) & \text{si } r(q) = q \\ q & \mapsto & 0 & \text{sinon.} \end{cases}$$

   Alors l'équation maîtresse réduite $\frac{dY(t)}{dt} = [P_r \circ A \circ Z_r](Y(t))$ définit, pour chaque état représentant de sa classe d'équivalence, l'évolution de la probabilité que le système soit dans un état symétrique à cet état au cours du temps.

Exemple 6.3.15 *Dans le cas d'étude présenté en section 6.1.1.1, les états potentiels du système sont notés* $q_{i,j,k,l}$ *où* i *est le nombre d'occurrences de l'unique configuration du monomère de la protéine A,* j *celui de la configuration du dimère de la protéine A dans laquelle les deux occurrences de la protéine A sont liées par leurs sites x,* k *celui de la configuration du dimère de la protéine A dans laquelle les deux occurrences de la protéine A sont liées par leurs sites y et* l *celui de la configuration du dimère de la protéine A dans laquelle les deux occurrences de la protéine A sont liées par le site x de l'une et le site y de l'autre.*

Les représentants des classes d'équivalence d'états sont choisis en assimilant toutes les occurrences de configurations du dimère à des occurrences de la configuration du dimère dans laquelle les deux occurrences de la protéine A sont liées par leurs sites x.

Les projections P_r *et* Z_r *sont alors définies de la manière suivante :*

$$P_r(P_t(q_{i,j,k,l})) = \begin{cases} \sum_{0 \le j' \le j} \sum_{0 \le k' \le j-j'} P_t(q_{i,j',k',j-j'-k'}) & \textit{si } k = 0 \textit{ et } l = 0 \\ 0 & \textit{si } k \neq 0 \textit{ ou } l \neq 0. \end{cases}$$

$$Z_r(P_t(q_{i,j,k,l})) = \begin{cases} P_t(q_{i,j,k,l}) & \textit{si } k = 0 \textit{ et } l = 0 \\ 0 & \textit{si } k \neq 0 \textit{ ou } l \neq 0. \end{cases}$$

L'équation $\frac{dY(t)}{dt} = [P_r \circ A \circ Z_r](Y(t))$ *définit l'équation maîtresse réduite suivante :*

$$\begin{cases} \frac{dP_t(Q_{6,0,0,0})}{dt} = 2K^d P_t(Q_{4,1,0,0}) - 30K P_t(Q_{6,0,0,0}) \\ \frac{dP_t(Q_{4,1,0,0})}{dt} = 30K P_t(Q_{6,0,0,0}) + 4K^d P_t(Q_{2,2,0,0}) - \left(12K + 2K^d\right) P_t(Q_{4,1,0,0}) \\ \frac{dP_t(Q_{2,2,0,0})}{dt} = 12K P_t(Q_{4,1,0,0}) + 6K^d P_t(Q_{0,3,0,0}) - \left(2K + 4K^d\right) P_t(Q_{2,2,0,0}) \\ \frac{dP_t(Q_{0,3,0,0})}{dt} = 2K P_t(Q_{2,2,0,0}) - 6K^d P_t(Q_{0,3,0,0}) \end{cases}$$

dans laquelle les états sont notés avec un Q majuscule pour les distinguer de ceux du système initial. Les solutions $X(t)$ *et* $Y(t)$ *de l'équation maîtresse initiale et de l'équation maîtresse réduite sont liées par la relation,* $Y(t) = P_r(X(t))$*, à tout instant de l'exécution du système.*

La relation de bisimulation avant suffit alors pour prouver les résultats constatés en figure 6.14(a) et en figure 6.14(e). Ainsi la distribution des états au cours du temps dans le modèle simplifié reste identique à celle obtenue dans le modèle initial en regroupant les états par classes d'équivalence. Il n'est pas nécessaire que la distribution initiale des états soit symétrique (voir en figure 6.14(e)).

6.3.3.2 Bisimulation en arrière

La propriété 6.3.2 assure que des invariants quantitatifs, prenant la forme de rapport de proportionnalité entre des probabilités des états symétriques, sont préservés lors de l'exécution stochastique d'un modèle symétrique.

Il faut dans un premier temps définir la notion de symétrie pour les distributions d'états. Une distribution d'états est une fonction de l'ensemble des états potentiels (qui peuvent être vus comme des graphes à sites qui ne peuvent être raffinés davantage sans ajouter de composante connexe ou encore comme des multi-ensembles de configurations d'espèces biochimiques) dans l'intervalle des nombres réels compris entre 0 et 1, dont la somme de toutes les images, pour tous les états potentiels du système est égal à 1. Une distribution d'états sera alors dîtes symétrique par rapport à un type d'échanges de sites, si et seulement si, pour toute paire d'états, telle que l'un des états puisse être obtenu à partir de l'autre en appliquant des échanges de ce type, les valeurs prises par les images des deux états par la distribution sont proportionnelles au nombre d'occurrence de ces états dans l'orbite de ces états pour ce type d'échanges de sites.

Exemple 6.3.16 *Dans l'exemple introduit dans la section 6.1.1.1, seuls les rapports entre les distributions des états qui comportent le même nombre d'occurrences de l'unique configuration du monomère de la protéine A et le même nombre global d'occurrences des trois configurations de son dimère, doivent être regardés. Le nombre d'occurrence d'un état dans l'orbite obtenue en échangeant les sites x et y dans des occurrences de la protéine A est inversement proportionnel à son nombre d'automorphisme. Soient deux états $q_{i,j,k,l}$ et $q_{i',j',k',l'}$, comportant respectivement i et i′ occurrences de l'unique configuration du monomère de la protéine A, j et j′ occurrences du dimère dans sa configuration dans laquelle la liaison porte sur les deux sites x des occurrences de la protéine A, k et k′ occurrences du dimère dans sa configuration dans laquelle la liaison est entre les deux sites y des occurrences de la protéine A, et enfin l et l′ occurrences du dimère de la protéine A dans sa configuration asymétrique. Ces états admettent respectivement $i!\cdot j!\cdot k!\cdot l!\cdot 2^j\cdot 2^k$ et $i'!\cdot j'!\cdot k'!\cdot l'!\cdot 2^{j'}\cdot 2^{k'}$ automorphismes. Dans les deux cas, les quatre premiers facteurs correspondent aux permutations entre les différentes occurrences d'une même configuration d'espèces biochimiques. Ensuite les deux suivants viennent du fait que chaque occurrence d'une configuration symétrique du dimère de la protéine A admet 2 automorphismes. En supposant que ces deux états sont symétriques pour l'échange des sites x et y dans des occurrences de la protéine A, il vient les deux contraintes supplémentaires $i = i'$ et $j + k + l = j' + k' + l'$. Ainsi $\frac{2^j\cdot 2^k}{2^{j'}\cdot 2^{k'}} = 2^{l'-l}$. Puis, le rapport entre les nombres d'automorphismes entre le premier et le second état est égal à $\frac{j!\cdot k!\cdot l!\cdot 2^{l'-l}}{j'!\cdot k'!\cdot l'!}$. Ainsi, une distribution d'états est symétrique, si pour chaque paire d'états $q_{i,j,k,l}$ et $q_{i',j',k',l'}$ tels que $i = i'$ et $j + k + l = j' + k' + l'$, le rapport entre la probabilité d'être dans l'état $q_{i,j,k,l}$ et celle d'être dans l'état $q_{i',j',k',l'}$ est égal à $\frac{j'!\cdot k'!\cdot l'!\cdot 2^{l-l'}}{j!\cdot k!\cdot l!}$.*

Les rapports de proportions observés en figure 6.15(a) correspondent bien à cette expression. Ainsi pour les états $q_{4,0,1,0}$ et $q_{4,1,0,0}$, ce rapport est égal à $\frac{1!\cdot 0!\cdot 0!\cdot 2^{0-0}}{0!\cdot 1!\cdot 0!}$, soit la valeur 1 ; pour les états $q_{4,0,0,1}$ et $q_{4,1,0,0}$, ce rapport est égal à $\frac{1!\cdot 0!\cdot 0!\cdot 2^{1-0}}{0!\cdot 0!\cdot 1!}$, soit la valeur 2 ; pour les états $q_{2,1,1,0}$ et $q_{2,0,0,2}$, ce rapport est égal à $\frac{0!\cdot 0!\cdot 2!\cdot 2^{0-2}}{1!\cdot 1!\cdot 0!}$, soit la valeur 2.

Cette propriété se décline sur plusieurs niveaux d'abstraction.

1. **Distribution de traces.** Partant d'une distribution symétrique d'états pour un type d'échanges de sites, si l'ensemble des règles du modèle est lui-même symétrique par rapport à ce type d'échanges de sites, alors les pas de calculs préservent la symétrie de la distribution des états. Plus précisément, étant donnée une distribution symétrique d'états, la distribution d'états obtenue après un pas de calcul sachant que ce pas de calcul appartient à une classe d'équivalence de pas de calculs est elle-aussi symétrique (quite à sommer, c'est une conséquence directe de la propriété 6.3.2). En tenant compte du temps et en procédant par induction sur le nombre de pas de calculs, il est possible d'étendre ce résultat aux distributions obtenues après un ensemble de traces [117]. Soit β la fonction d'abstraction définie sur les traces, qui consiste à remplacer chaque état par sa classe d'équivalence et chaque pas de calcul par la classe d'équivalence pour les échanges de sites étudiés tout en préservant les intervalles pour les délais entre les pas de calculs. L'image d'une trace du modèle initial par la fonction β est alors appelée une trace abstraite. Si à la

model	sites	rules	species		reactions	
			original	reduced	original	reduced
kinase/phosphatase	n	$6n$	$2+4^n$	$2+\binom{n+3}{3}$	$6n4^{n-1}$	$2n\binom{n+2}{2}$
multiple phosphorylation	n	$n2^n$	2^n	$n+1$	$n2^n$	$2n$
mult. phosphoryl. with counter	n	$2n^2$	2^n	$n+1$	$n2^n$	$2n$

Figure 6.30: Principales caractéristiques combinatoires des différents modèles testés en fonction du nombre de site n par occurrences de la protéine.

fois l'ensemble des règles et la distribution initiale des états sont symétriques pour le type d'échanges de sites en question, alors pour tout ensemble $X^\sharp$ de traces abstraites, la distribution conditionnelle des états obtenue après avoir appliqué une trace sachant que son abstraction est dans l'ensemble $X^\sharp$ est elle-même symétrique.

2. **Èquation maîtresse.** Il est possible d'exploiter la propriété 6.3.2 directement sur l'équation maîtresse [22]. Ainsi si un ensemble de règles est symétrique par rapport à un type d'échanges de sites et si la distribution initiale des états est symétrique par rapport à ce même type d'échanges de sites, alors, par sommation, la distribution des états à l'instant t est elle aussi symétrique par rapport à ce type d'échanges de sites.

Exemple 6.3.17 *Dans l'exemple introduit dans la section 6.1.1.1, cette propriété justifie les observations faites en figure 6.15(a). Par ailleurs, en figure 6.15(e), est donné un exemple de simulation numérique dans lequel l'ensemble des règles est symétrique sans que la distribution initiale des états ne le soit. Dans ce cas, la distribution des états convergent vers une distribution symétrique. C'est en fait le cas, car tout état accessible du système appartient à une unique composante fortement connexe maximale, elle-même stable par symétrie. La distribution des états restreintes à chaque composante fortement connexe terminale converge vers une distribution ergodique.*

Tout comme c'était le cas pour la sémantique différentielle, il est possible d'utiliser les symétries par rapport à un type d'échanges de sites pour éliminer des variables dans l'équation maîtresse. Il faut pour cela, choisir un représentant pour chaque classe d'équivalence d'états. Le représentant d'un état q est notée $r(q)$. Le nombre d'automorphismes de q est alors noté $aut(q)$.

Les deux fonctions de l'ensemble des distributions d'états vers lui-même sont définies ci-dessous :

$$Eq_r(d) : \left\{ q \mapsto \frac{aut(r(q))}{aut(q)} \cdot d(r(q)) \right. \qquad Z_r(d) : \left\{ \begin{array}{lll} \mathcal{V} & \to & \mathbb{R} \\ q & \mapsto & \rho(q) \quad \text{si } r(q) = q \\ q & \mapsto & 0 \quad \text{sinon.} \end{array} \right.$$

L'élimination des états qui ne sont pas les représentants de leurs classes d'équivalence est alors formalisée par l'équation suivante :

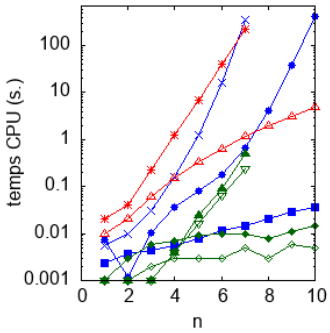
$$[[Z_r \circ Eq_r] \circ [\mathbb{F}]](d) = [[Z_r \circ \mathbb{F} \circ Eq_r] \circ [Z_r \circ Eq_r]](d)$$

qui est réalisée dès lors que l'ensemble des règles et la distribution d'états d sont tous deux symétriques par rapport aux échanges de sites en question.

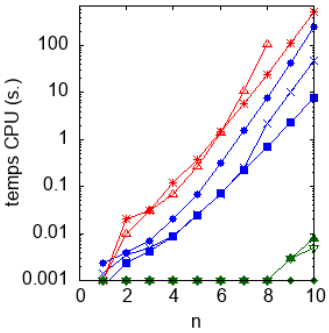
6.4 Étude de performance

Cette approche est implantée dans un outils appelé KaDe. Dans [29], est effectuée une étude de performance de cet outils, à la fois en terme de capacité de réduction et en terme de temps de calcul, ainsi qu'une comparaison avec d'autres outils existants pour réduire les sémantiques différentielles et stochastiques des réseaux réactionnels. Cette étude de performance mériterait d'être faite à nouveau avec les versions récentes des outils cités. Par ailleurs, il serait bien de comparer avec l'outils Jupiter [33] qui opère directement sur les systèmes d'équations différentielles.

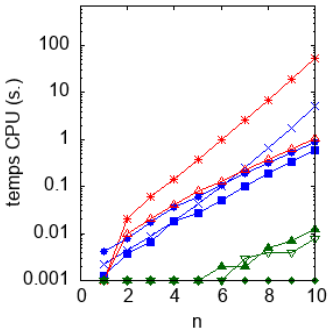
En particulier, trois modèles paramétriques ont été considérés.



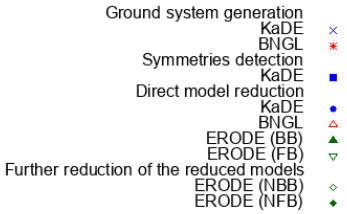
(a) Modèle avec kinase et phosphatase.



(b) Modèle de phosphorylations multiples.



(c) Modèle de phosphorylations multiples implanté par un compteur.



(d) Légende.

Figure 6.31: Comparaison entre les performances en temps de calcul des outils KaDe, BNGL, et ERODE, sur un MacBookPro avec un puce `Intel Core i7` cadencée à `2,8 GHz` CPU et une mémoire `DDR` de `16 Go 1600 MHz`. Les exécutions sont interrompues lorsque leurs temps d'exécution dépasse 10 minutes. Ces tests ont été effectué en 2017 (il faudrait les mettre à jour). ***Génération du réseaux réactionnels non-réduit*** : sont donnés le temps CPU pour la génération du réseau non réduit avec les outils KaDe et BNGL. ***Inférence des symétries*** : est tracé le temp total pour calculer les paires de sites équivalents avec KaDe. ***Réduction de modèle*** : sont reportés les temps de calculs pour obtenir le réseau réactionnel réduit avec KaDe, BNGL, et ERODE. Pour KaDe le point d'entrée est l'ensemble des règles d'interaction des modèles, alors que pour les deux autres c'est le réseau réactionnel non réduit. ***Réduction supplémentaire*** : est donné le temps CPU pour l'exécution d'ERODE sur les réseaux réactionnels réduits par KaDe.

- La première famille implique une kinase, une phosphatase et une protéine cible. La protéine cible dispose de n sites (n étant un paramètre entier), pouvant être phosphorylés ou non, et liés ou non à la kinase ou à la phosphatase. Une occurrence de la kinase peut se lier ou se séparer de chaque site d'une occurrence de la protéine cible dans une configuration où ce site est non phosphorylé et libre. Dans ce cas, l'occurrence de la kinase peut également phosphoryler l'occurrence de ce site tout en se dissociant de celle-ci. Réciproquement, une occurrence de la phosphatase peut se lier et de séparer de chaque site d'une occurrence de la protéine cible dans une configuration où ce site est phosphorylé et libre. Dans ce cas, l'occurrence de la kinase peut également déphosphoryler l'occurrence de ce site tout en se dissociant de celle-ci.

- La seconde famille est inspirée du comportement de la protéine *Kai*. Cette protéine joue un rôle crucial dans la contrôle des oscillations du cycle circadien. Elle implique une protéine avec n sites de phosphorylation. La kinase et la phosphatase ne sont pas décrites explicitement. Les taux de phosphorylation et de déphosphorylation de chaque occurrence de chaque site est supposé dépendre du nombre de sites déjà phosphorylé dans l'occurrence de la protéine correspondante.

- La troisième famille représente le même système que la seconde famille, mais en utilisant une astuce suggérée par Pierre Boutillier pour limiter le nombre de règles à écrire pour décrire le comportement des occurrences de configurations de la protéine. Cet astuce consiste à ajouter un site fictif à chaque protéine qui sera lié à une chaîne d'occurrences de protéines fictives, en maintenant l'invariant que la longueur de la chaîne des occurrences de la protéine fictive est toujours égale au nombre de sites phosphorylés dans la

configuration de la protéine en question. Ceci permet de connaître le nombre de sites phosphorylés dans une configuration de la protéine, sans avoir à énumérer les différentes possibilités pour le sous-ensemble des sites phosphorylés. Ce mécanisme est d'ailleurs à la base de l'implantation des protéines à compteur dans Kappa [16]. Il reprends des comportements présent dans la nature, notamment ceux impliqués dans le processus d'ubiquitination.

D'autres examples, notamment la plupart de ceux de la base de tests du langage BNGL sont donnée en information complémentaire [28].

Pour chaque jeu de paramètres des modèles, sont donnés, en figure 6.30, les nombres de règles, de configurations d'espèces biochimiques et de réactions, ainsi que le nombre de configurations d'espèces biochimiques et le nombre de réactions dans la modèle réduit correspondant. Les tests sont fait en faisant varier le paramètre n entre 1 et 10. Les performances des outils sont tracées en figure 6.31. En particulier sont comparés les temps de génération du réseau réactionnel réduit obtenu avec KaDe et avec BNGL. Il est important de remarquer que dans l'approche utilisant BNGL, les symétries entre les sites doivent être spécifier par l'utilisateur, alors qu'elles sont inférées automatiquement par KaDe. Les réseaux réduits obtenus sont les mêmes pour tous les modèles de cette batterie de tests. Cependant, le temps de génération avec l'outils KaDe est bien meilleur.

Enfin, les performances de KaDe ont été comparée avec celle d'ERODE. ERODE adopte une approche algébrique pour inférer des bisimulation, en-avant ou en arrière, dans des réseaux réactionnels. Ont été considérées à la fois la version rapide [35] et celle complète [34] (qui calcule les meilleurs bisimulations). L'outils ERODE a été testé à la fois sur les réseaux réactionnels initiaux et ceux réduit. Dans notre batterie de test (qui inclue ceux de la plate-forme BNGL), aucune nouvelle réduction n'a été trouvée, même avec la version complète. Ceci montre la prépondérance des symétries dans les tests réalisés. L'approche utilisant KaDe est plus rapide que la version rapide de l'outils ERODE. Cela vient du fait que KaDe opère directement sur la structure des règles, et ne nécessite pas de considérer l'ensemble des réactions. Par ailleurs, l'utilisation d'ERODE impose d'avoir une description extensionnelle du réseau réactionnel ou de sa sémantique différentielle, ce qui est prohibitif pour les modèles de grandes ou moyennes tailles.

6.5 Pour aller plus loin

Dans ce chapitre a été esquissé une méthode pour utiliser des symétries entre paires de sites d'interaction pour réduire la dimension de la sémantique différentielle et la taille de la sémantique stochastique des modèles de réécriture de graphes à sites. Cette approche a été implantée dans l'outils KaDe.

Cette méthode a été obtenue en étendant la définition des échanges de sites dans un motif, d'une part, aux plongements, aux règles et aux raffinements de règles, d'autres parts aux états de la sémantique différentielle et aux états de la sémantique stochastique. Il en résulte une définition des états symétriques (dans la sémantique différentielle et dans la sémantiques stochastique) et des ensembles symétriques de règles. Les symétries entre paires de sites induisent une bisimulation avant-arrière. Ainsi dans un modèle issu d'un ensemble de règle symétrique pour les échanges de sites dans les occurrences d'une protéine, l'exécution du modèle dans la sémantique différentielle et celle dans la sémantique stochastique peuvent être vues à échanges de ces sites près dans les configurations d'espèces biochimiques. De plus, lorsque l'état initial — dans la sémantique différentielle — ou la distribution initiale des états — dans la sémantique stochastique — est symétrique, alors cela reste vrai au cours de l'exécution du système.

Combinaion de réduction de modèles. Comme montré dans [27], les réductions de modèles décrites dans le chapitre 4 et dans le chapitre 5 peuvent se combiner avec les réductions basées sur les échanges de sites symétriques. Le but est d'alors d'oublier au moins autant d'information que dans chacune des deux réductions, et parmi toutes les réductions qui oublient au moins autant d'information, choisir celle qui en oublie le moins. Pour la sémantique différentielle, la principale difficulté vient du fait que les réductions de modèles du chapitre 4 ne sont pas engendrées par des relations d'équivalence sur l'ensemble des configurations des espèces biochimiques. En effet, une configuration d'espèces biochimiques donnée peut apparaître dans plusieurs combinaisons linéaires. Le diagramme dessiné en figure 6.32 propose une manière de composer une réduction classique avec une réduction basée sur une relation d'équivalence sur les configurations des espèces biochimiques. Dans ce dernier, outre les diagrammes commutatifs intermédiaires qui correspondent à la réduction de modèle classique et à la bisimulation en avant induite par la relation d'équivalence r, se trouvent deux nouveaux diagrammes intermédiaires. Ceux-ci permettent d'étendre la notion de symétries sur les variables du modèle réduit du chapitre 4. Pour cela, il

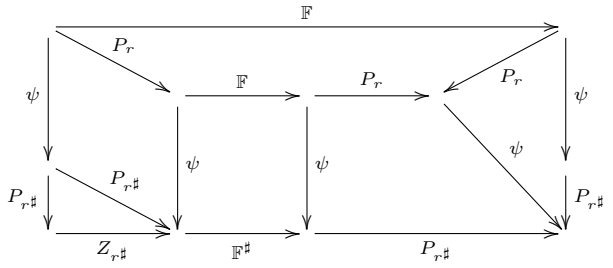
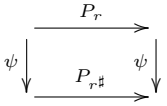


Figure 6.32: Diagramme commutatif pour composer une réduction de modèle classique et une réduction basée sur des échanges de paires de sites symétriques. Dans celui-ci, $\mathbb{F}$ désigne la dynamique du système dans la sémantique concrête, $\mathbb{F}^\sharp$ sa contre-partie abstraite dans le modèle obtenu par la réduction classique et ψ la fonction d'abstraction correspondante. Le fonction r spécifie un choix de représentantes pour les classes d'équivalence des configurations des espèces biochimiques et la fonction $r^\sharp$ un choix de représentantes pour les classes d'équivalence des variables du modèle obtenu par la réduction classique. P_r et $P_{r^\sharp}$ sont les fonctions de projection qui regroupent la quantité des éléments sur leurs classes d'équivalence, alors que la fonction $Z_{r^\sharp}$ annule la quantité des variables du modèle réduit qui ne sont pas les représentantes de leurs classes d'équivalence. Les deux fonctions pour choisir les représentantes doivent vérifier la contrainte $P_{r^\sharp} \circ \psi = \psi \circ P_r$. Sous cette hypothèse, le modèle initial peut être réduit par la fonction d'abstraction $P_{r^\sharp} \circ \psi$. La contre-partie abstraite de la dynamique du système est alors donnée par la fonction $P_{r^\sharp} \circ \mathbb{F}^\sharp \circ Z_{r^\sharp}$.

faut choisir une fonction de représentantes $r^\sharp$ pour les classes d'équivalence modulo les symétries des variables du modèle obtenu par la réduction classique. Le premier nouveau diagramme exploite le fait que la fonction d'oublie $Z_{r^\sharp}$ qui annule la quantité des variables du modèle obtenu par la réduction classique qui ne sont pas les représentantes de leurs classes d'équivalence, et la fonction d'abstraction $P_{r^\sharp}$ qui regroupe ces quantités sur les représentantes des classes d'équivalence, sont deux projection linéaires définies sur le même espace et partageant la même image. De ce fait, $P_{r^\sharp} = Z_{r^\sharp} \circ P_{r^\sharp}$. Le choix des représentantes pour les configurations des espèces biochimiques, celui des représentantes pour les variables du modèle réduit et la fonction d'abstraction entre le modèle initial et le modèle réduit (avant la prise en compte des symétries) doivent respecter le second nouveau diagramme commutatif :



qui signifie qu'étant donné un état concret, abstraire cet état puis regrouper les quantités par classes d'équivalence des variables du modèle réduit ou regrouper les quantités par classes d'équivalence de configurations d'espèces biochimiques, puis abstraire le résultat, donnent la même fonction. Cette condition signifie qu'aucune corrélation entre les états de deux sites symétriques n'est abstrait par la réduction de modèles, ce qui revient à dire tout motif d'intérêt du modèle obtenu par la réduction classique qui contient un site, contient tous les sites qui lui sont symétriques. Sous cette hypothèse, la combinaison des deux réductions de modèles donne lieu à une nouvelle. La fonction d'abstraction applique la fonction d'abstraction de la réduction de modèle du chapitre 4, puis regroupe les quantités de motifs d'intérêt sur les représentants de leurs classes d'équivalence. La contre-partie abstraite de la fonction $\mathbb{F}$ s'obtient en éliminant la contribution des motifs d'intérêt qui ne sont pas les représentants de leurs classes d'équivalence, puis en applicant la contre-partie abstraite de la réduction de modèle du chapitre 4, puis en regroupant les quantités de motifs d'intérêt sur les représentants de leurs classes d'équivalence.

Les combinaisons de réductions de modèles dans le cadre stochastique prennent la forme de compositions et de sommes amalgamées, dans la catégorie des relations d'équivalence. De ce fait, le diagramme donné en figure 6.33 donne une hiérarchie des sémantiques de traces pour les modèles de réécriture de graphes à sites. De gauche à droite, l'abstraction oublie graduellement de l'information sur les occurrences de protéines. La sémantique individuelle définit une distribution de traces sur des états dans lesquels les occurrences de protéines portent un identifiant unique qui les suit tout le long des exécutions potentielles du système. Les sémantiques de population, elles, décrivent l'état du système à permutation des identifiants près, c'est à dire à isomorphisme près. Le terme population vient du fait qu'une classe d'équivalence d'un état pour les isomorphismes contient

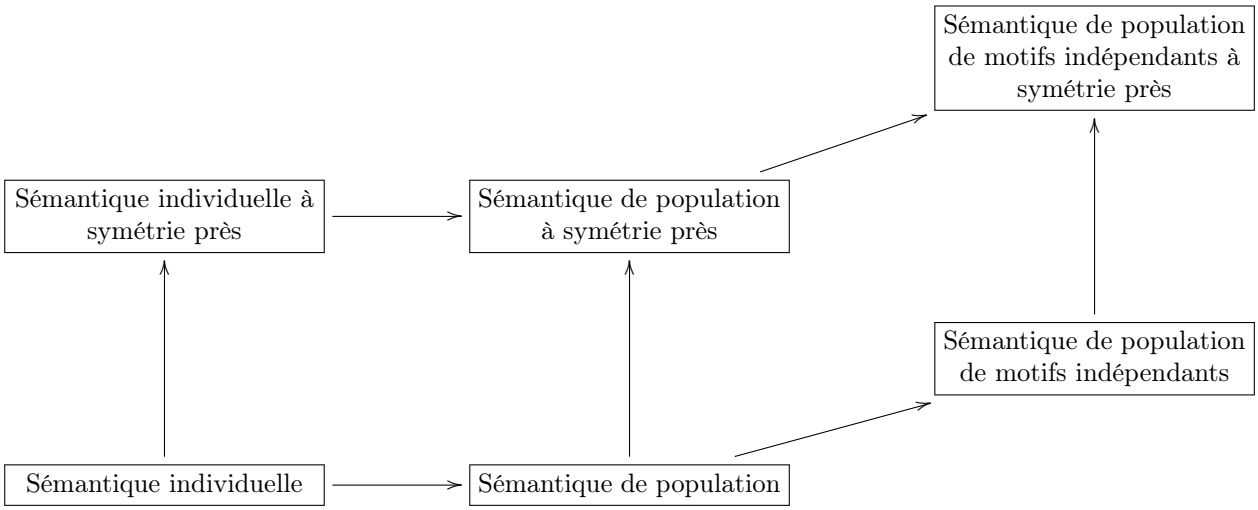
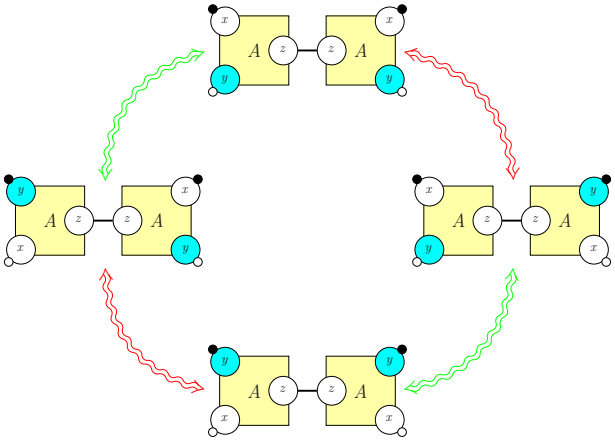


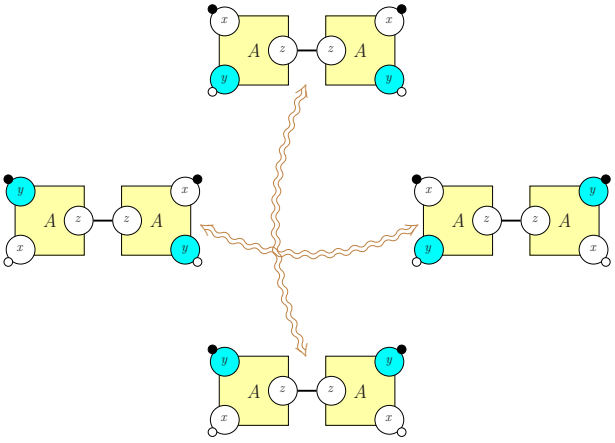
Figure 6.33: Hiérarchie de sémantiques de traces dans le cadre stochastique. Les sémantiques individuelles décrivent des distributions de traces portant sur des occurrences de protéines annotées par un identifiant unique qui les suit le long de l'exécution du modèle. Les sémantiques de population sont obtenues en raisonnant sur les graphes à sites à isomorphisme près, ce qui revient à oublier ces identifiants. Enfin, les sémantiques de motifs indépendants sont obtenus en découpant les occurrences de configurations de protéines selon la méthode présentée en chapitre 5. Chaque sémantique existe sous deux formes selon que les éventuelles symétries entre les paires de sites aient été utilisées pour réduire le modèle ou non. Chaque flèche représente un quotient par une relation d'équivalence sur les ensemble des états de la sémantique à la source de cette flèche. Deux relations se composent séquentiellement, lorsque que la seconde est plus grossière que la première. Deux relations peuvent aussi se combiner en utilisant une somme amalgamée. La relation obtenue est alors la plus petite relation qui identifie au moins toute paire d'éléments identifiées dans au moins l'une ou l'autre des deux relations initiales. Ce diagramme est valide pour les bisimulations avant-arrière, pour les bisimulation en avant et pour les bisimulations en arrière.

peut être décrite comme un multi-ensemble d'occurrences de configurations d'espèces biochimiques. Il est alors possible d'aller encore plus loin en découpant les occurrences de protéines sur les parties qui se comportent de manière indépendante. Les flèches verticales correspondent, elles, au quotient des états par les échanges de sites symétriques. La somme amalgamée de deux relations d'équivalence se définit comme la relation d'équivalence la plus grossière qui est au moins aussi fine que les deux relations données en argument. Elle se calcule en effectuant la clôture transitive du graphe formé par l'union des deux relations. Cette opération est définie modulo le choix des noms qui sont associés aux classes d'équivalence dans la somme amalgamée des deux relations. Grâce à cela, il n'y a pas d'ordre à fixer entre les deux relations. Cette hiérarchie est valable pour les bisimulations avant-arrière, pour les bisimulations en avant et pour les bisimulations en arrière. Dans le dernier cas, il est intéressant de remarquer que toutes les flèches d'abstraction partant d'une sémantique individuelle correspondent à des bisimulations uniformes (dans lesquelles deux objets en relation ont le même poids). Il est possible de retrouver les poids qui interviennent dans le quotient des sémantiques de populations en définissant ces quotients comme la restriction de deux quotients de la sémantique individuelle. Le poids de chaque élément correspond alors au rapport entre le cardinal de ses classes d'équivalence. Ceci donne une autre manière de retrouver les rapports de proportionnalités en inverse du nombre d'automorphismes.

D'autres groupes de symétries. Dans [73] est présenté un cadre générique pour suivre les sémantiques des modèles de règles de réécriture modulo des groupes de symétries. Les échanges de sites qui ont été étudiés dans ce chapitre en sont un cas particulier. Dans ce cadre plus général, il est possible de définir des sous-groupes de symétries à partir de groupes existants. Par exemple, il est possible de considérer les configurations à permutation circulaire près de l'état d'un sous-ensemble de ses sites. Il est aussi possible de contraindre les échanges de sites qui sont appliquées à deux occurrences de configurations d'une protéine dans une même composante connexe. La question essentielle est de comprendre si de tels sous-groupes de symétries engendrent également des bisimulations en avant et des bisimulations en arrière sur les ensembles symétriques de règles.



(a) Échanges hétérogènes de sites.



(b) Échanges homogènes de sites.

Figure 6.34: Orbites des configurations du dimère de la protéine A dans lesquelles exactement un site par occurrence est phosphorylée. En figure 6.34(a), pour les échanges hétérogènes de sites, toutes les configurations appartiennent à la même orbite. En figure 6.34(b), les échanges homogènes de sites séparent les dimères dont les deux occurrences de la protéine A sont dans la même configuration, des autres.

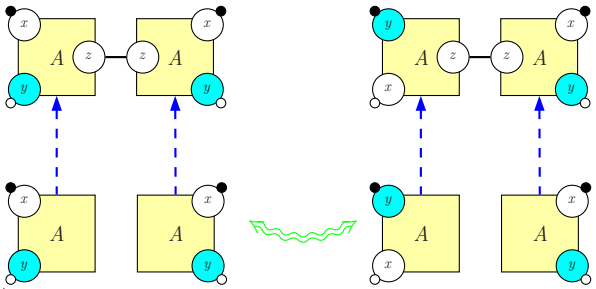
Exemple 6.5.1 *Dans cet exemple est considéré deux groupes de symétries qui permettent d'échanger deux sites dans les occurrences d'une protéine. Dans le premier, les échanges peuvent être choisis indépendamment pour chaque occurrence de la protéine (c'est la notion de symétrie qui a été étudiée dans ce chapitre). Ces échanges de sites seront dits hétérogènes. Dans le second, le même échange de sites doit être appliqué à toutes les occurrences de la protéine dès lors qu'elles sont dans la même composante connexe. Ces échanges de sites seront dit homogènes.*

Soit A une protéine comportant trois sites : deux sites de phosphorylation x et y et un site de liaison z. Les sites z de deux occurrences de la protéine A peuvent se lier entre eux. Une attention particulière est portée sur l'ensemble des configurations du dimère de la protéine dans lesquelles exactement un site par occurrence de la protéine, est phosphorylé. En figure 6.34(a), cet ensemble forme une orbite pour les échanges hétérogènes des sites x et y dans les occurrences de la protéine A (cette orbite est constituée de 4 éléments dont deux sont isomorphes). En figure 6.34(b), cet ensemble forme deux orbites disjointes pour les échanges homogènes de sites, selon que les sites phosphorylés dans les deux occurrences de la protéine soient les même ou non.

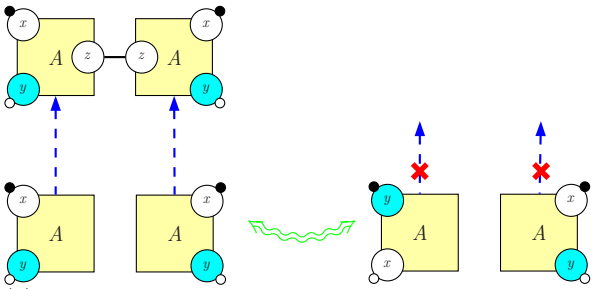
C'est un point essentiel pour étendre l'utilisation du langage Kappa à d'autres champs d'applications, comme la chemioinformatique. Dans ce domaine, les nœuds des graphes à sites représentent des atomes et les sites d'interaction des électrons susceptibles d'être partagés ou liés deux à deux. Deux différences principales entre les modèles en biochimie et ceux en chemioinformatique sont les suivantes. En chemioinformatique, les atomes ont peu d'électrons disponibles. Par contre, des symétries complexes sont à prendre en compte. Ainsi en stéréochimie, seules les permutations de sites de signe paire (celles qui sont engendrées par un nombre paire d'échanges de sites deux à deux) peuvent être appliquées aux occurrences des atomes. Par ailleurs, il est impossible de dissocier deux électrons qui participent à une double liaison. Enfin, lorsque deux électrons participants à une double liaison sont échangés l'un à l'autre dans l'occurrence d'un atome, il doit en être de même pour les deux électrons de l'autre côté de la liaison. Toutes ces contraintes se traduisent sous la forme d'un sous-groupe de symétries. L'étude et la prise en compte de ce groupe de symétries permettront de compléter les outils offert pour la chemioinformatique par le language mød [4] avec des analyses statiques, des analyses causales et des réductions de modèles.

En toute généralité, le cadre algébrique considère des groupes de transformations sur les graphes à sites telles que les transformations qui s'appliquent aux motifs cibles des plongements puissent être restreintes à leurs motifs sources. Par contre, contrairement aux échanges de sites qui ont été proposés dans ce chapitre, il n'est pas toujours possible d'étendre une transformation qui s'applique au motif source d'un plongement à son motif cible. Cette propriété ne passe pas au sous-groupe.

Exemple 6.5.2 *Pour poursuivre l'exemple précédent, est donné en figure 6.35 un exemple d'échange de sites*



(a) Lorsque les échanges de sites sont indépendants, il est possible d'étendre l'échange des états des sites x et y dans la première occurrence de la protéine A dans le motif source du plongement au motif cible.



(b) C'est impossible lorsque le même échange doit être effectué sur toutes les occurrences d'une même protéine au sein des composantes connexes.

Figure 6.35: Un exemple de plongement, pour lequel un échange de sites opérant sur le motif source peut s'étendre au motif cible uniquement pour le cas des échanges de sites indépendants. Le motif source est constitué de deux occurrences de la protéine A, toutes deux dans une configuration dans laquelle le site x est phosphorylé, mais pas le site y. L'état du site z n'est pas précisé. Le plongement consiste à ajouter le site z dans les deux occurences de la protéine et à préciser que ces sites sont liées. Dans le motif cible, les deux occurrences de la protéine A sont donc dans la même composante connexe. L'échange de sites en question consiste à intervertir les états des sites x et y dans la première occurence de la protéine A, mais pas dans la deuxième. La question est de savoir si un tel échange de sites peut s'étendre au motif cible du plongement. Avec le groupe des échanges hétérogènes de sites (voir en figure 6.35(a)), cela ne pose pas de problèmes, puisque les choix d'échanger les états des sites x et y dans chaque occurrence de la protéine A sont indépendants. En revanche, avec le sous-groupe des échanges homogènes de sites (voir en figure 6.35(b)), ce n'est pas possible. Comme les deux occurrences de la protéine A sont dans la même composante connexe dans le motif cible, il faut soit ne procéder à aucun échange, soit échanger les états des sites x et y dans les deux occurrences de la protéine A simultanément.

qui s'applique au motif source d'un plongement, qui peut s'étendre au motif cible dans le groupe des échanges hétérogènes de sites, mais pas dans celui des échanges homogènes de sites.

Le motif source est constitué de deux occurrences de la protéine A, toutes deux dans une configuration dans laquelle le site x est phosphorylé, mais pas le site y. Les états de leurs sites z respectifs ne sont pas décrits. Le plongement ajoute ces deux sites tout en spécifiant qu'ils sont liés. L'exemple consiste à essayer d'étendre l'échange entre les sites x et y dans la première occurrence du motif source au motif cible du plongement. Ceci n'est possible que pour le groupe des échanges hétérogènes de sites. En effet, la difficulté vient du fait que les deux occurrences de la protéine A appartiennent à deux composantes connexes différentes dans le motif source du plongement, mais à la même composante connexe dans le motif cible. Avec le groupe des échanges de sites hétérogènes, ce n'est pas un soucis, puisque les choix d'échanger les états des sites x et y dans chacune des occurrences de la protéine A peuvent être faits de manière indépendante même dans le motif cible. Par contre, avec le groupe des échanges homogènes de sites, c'est tout simplement impossible puisqu'échanger les états des sites x et y dans la première occurrence de la protéine A sans échanger ceux-ci dans la seconde, n'est pas autorisé.

En conséquence, la propriété fondamentale (voir section 6.3.1.4, page 123) n'est pas réalisée pour tous les sous-groupes de symétries : les transformations ne peuvent pas toujours être propagées de gauche à droite ou de droite à gauche le long des pas de calculs. En effet, étant donnée une transformation qui s'applique au membre gauche d'un pas de calcul, celle-ci peut être restreinte à la partie commune entre le membre gauche et le membre droit du pas de calcul. Il s'agit juste de la restriction d'une transformation qui s'applique au motif cible d'un plongement à son motif source. Par contre, il n'est pas toujours possible d'étendre la transformation obtenue au membre droit du pas de calcul. Ainsi, même dans le cas d'un ensemble symétrique de règles pour un sous-groupe de transformations, les symétries correspondantes ne définissent pas toujours une bisimulation en avant. De la même manière, étant donnée une transformation qui s'applique au membre droit d'un pas de calcul, celle-ci peut être restreinte à la partie commune entre le membre gauche et le membre droit du pas de calcul, mais rien ne garanti en général qu'il soit possible d'étendre la transformation obtenue au membre gauche du pas de calcul. Ainsi, même dans le cas d'un ensemble symétrique de règles pour un sous-groupe de transformations, les symétries correspondantes ne définissent pas toujours une bisimulation en arrière.

Une question essentielle en suspens est d'identifier des sous-groupes d'échanges de sites intéressants, et pour

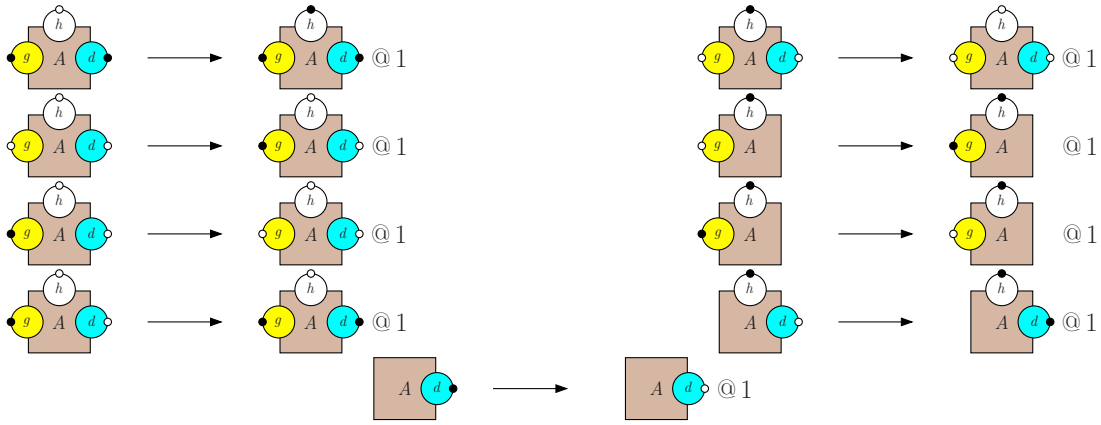


Figure 6.36: Un ensemble de règles dans lequel deux sites d'une protéine ont le même comportement dans les configurations de celle-ci dans lesquelles un troisième site est phophosrylé.

chacun de caractériser les ensembles symétriques de règles pour lesquels ils engendrent des bisimulations en avant, des bisimulations en arrière ou des bisimulations avant-arrière.

Symétries contextuelles. Un autre exemple de sous-groupes de transformations important est celui des symétries contextuelles entre paires de sites. Elles apparaissent lorsque plusieurs sites ont des comportements identiques mais uniquement dans certaines configurations de leur protéine. Il est alors possible de spécifier un groupe de transformations pour échanger l'états de ces sites uniquement dans les configurations en question. Il reste à savoir si ce groupe induit une bisimulation avant-arrière, juste une bisimulation avant, juste une bisimulation arrière ou aucune bisimulation.

Exemple 6.5.3 *Le modèle donné en figure 6.36 comporte des symétries contextuelles. Il est constitué d'une seule protéine A qui comporte trois sites de phosphorylation. Chacun de ses trois sites peut se phosphoryler ou se déphosphoryler dans les occurrences des configurations de la protéine A sous certaines conditions sur l'état des autres sites. Les deux règles de la première ligne spécifient que le site h d'une occurrence de la protéine A peut se phosphoryler si elle est dans une configuration dans laquelle les deux sites g et d le sont et se déphosphoryler uniquement lorsqu'elle est dans sa configuration dans laquelle les deux sites g et d sont eux même non phosphorylés. En deuxième ligne, le site g d'une occurrence de la protéine A peut se phosphoryler si elle est dans une configuration dans laquelle les deux sites h et d sont non phosphorylés ou dans une configuration dans laquelle au moins le site h est phosphorylé. Ces deux conditions sont incompatibles, ce qui assure que deux règles ne peuvent jamais s'appliquer toutes les deux à la même occurrence d'une configuration de la protéine A. En troisième ligne, la déphosphorylation du site g est réalisée exactement sous les mêmes conditions que sa phosphorylation. En quatrième ligne, le site d d'une occurrence de la protéine A peut se phosphoryler si elle est dans une configuration dans laquelle soit le site h est non phosphorylé et le site g est phosphorylé, soit le site h est phosphorylé peu importe l'état du site g. En cinquième ligne, le site d d'une occurrence de la protéine A peut se faire déphosphoryler sans conditions. Les constantes d'interaction de chaque règle sont toutes fixées à* 1.

Pour mieux appréhender le comportement des occurrences de la protéine A, un système de transitions est donné en figure 6.37 pour décrire leurs exécutions potentielles. Il s'agit d'une trace locale (voir page 40). Il apparaît que les occurrences des protéines fonctionnent selon deux modes selon que le site h soit phosphorylé ou non. Dans la partie basse du système de transitions, lorsque le site h n'est pas phosphorylé, les phosphorylations des sites g et d se font de manière séquentielle, tout comme leurs déphosphorylations mais dans l'ordre inverse. À l'opposé, dans la partie haute de ce système, lorsque le site h est phosphorylé, les phosphorylations des sites g et d sont indépendantes, de même que leurs déphosphorylations. Ainsi les sites g et d ont le même comportement uniquement dans les occurrences de configurations de la protéine A dans laquelle le site h est phosphorylé.

Dans ce modèle, il est possible de regrouper les occurrences de protéine dans les configurations dans lesquelles le site h est phosphorylé, ainsi qu'exactement un des deux autres sites. Ainsi les deux configurations suivantes :

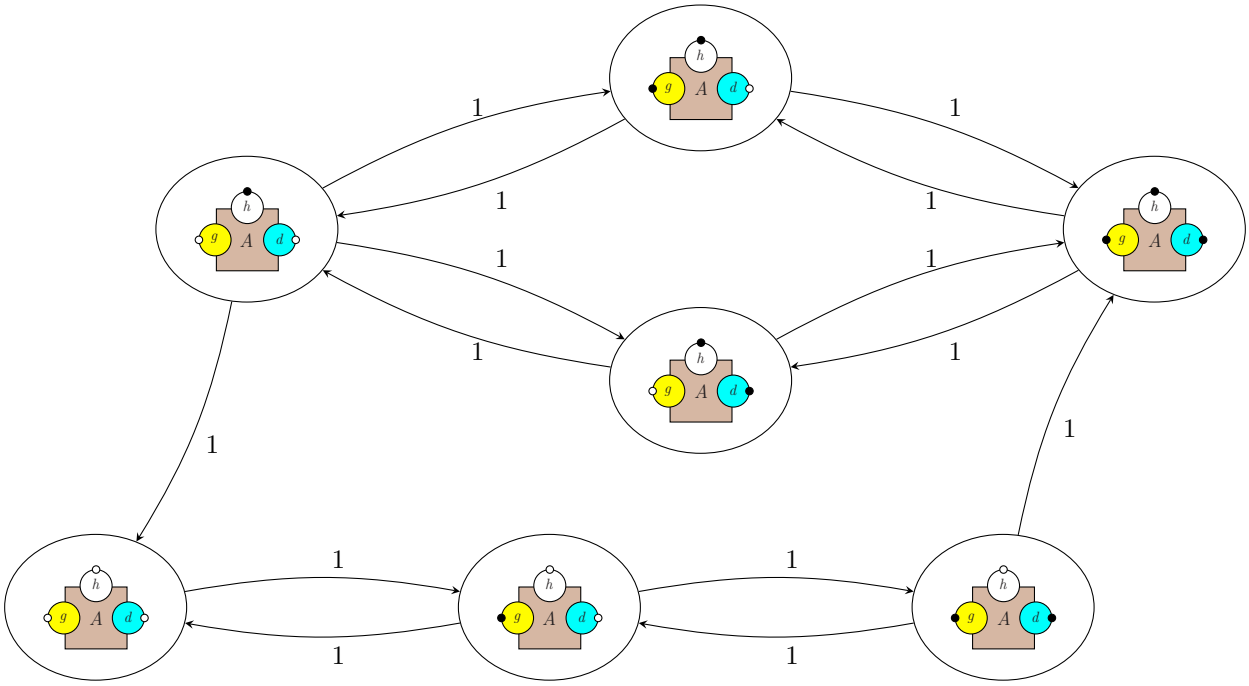


Figure 6.37: Le système de transitions qui décrit les évolutions potentielles de la configuration d'une protéine. Dans la partie basse, lorsque le site h n'est pas phosphorylé, le site g est toujours phosphorylé avant le site d. Ils ne sont donc pas symétriques dans ce contexte. Dans la partie haute, lorsque le site h est phosphorylé, les sites g et d se phosphorylent et se déphosphorylent selon les même propensité. Ils sont indiscernables dans ce contexte. La question est de savoir si cette information permet de définir une relation de bisimulation avant-arrière, juste avant, juste arrière, ou ni l'une ni l'autre, en assimilant la configuration de la protéine dans laquelle les sites h et g uniquement sont phosphorylées et celle dans laquelle uniquement les sites h et d sont phosphorylées. Comme les règles ont toutes 1 comme constante cynétique et qu'il n'y a pas de conflits entre elles (une transition est toujours le raffinement d'au plus une règle), chaque transition est de propensité 1.

sont symétriques et forment une classe d'équivalence, mais pas les deux configurations suivantes :

qui forment deux classes d'équivalence distinctes.

Cette relation d'équivalence engendre une bisimulation avant-arrière. Ceci tient essentiellement du fait que tout changement de l'état du site h dans une occurrence de la protéine A ne peut se faire que dans une configuration dans laquelle l'état des deux autres sites est le même.

Pour comprendre si des symétries contextuelles engendrent une bisimulation en avant, il faut examiner sous quelles conditions il est possible de casser le contexte dans lequel cette symétrie opère. Il faut, en effet, pouvoir déplacer les symétries de l'état juste avant l'application d'une règle d'interaction à l'état obtenu juste après avoir appliqué celle-ci. Comme les configurations de protéines qui ne satisfont les conditions contextuelles sont réduites à des singletons, cela revient à dire que l'application d'une règle d'interaction à deux configurations symétriques de protéines doit produire la même configuration de protéines, dès lors que cette règle brise la condition contextuelle.

Exemple 6.5.4 *Dans l'exemple des figures 6.36 et 6.37, s'il était possible de déphosphoryler le site h des occurrences de la protéine A sans condition, alors les deux configurations suivantes:*

n'auraient plus le même comportement vis à vis des classes d'équivalence de la relation (car la déphosphorylation du site h dans ces configurations donneraient deux configurations qui ne seraient pas dans la même classe d'équivalence). Celle-ci ne pourrait alors pas engendrer une bisimulation en avant.

Réciproquement, pour comprendre si des symétries contextuelles engendrent une bisimulation en arrière, il faut examiner sous quelles conditions il est possible de former le contexte dans lequel cette symétrie opère. Il faut, en effet, pouvoir déplacer les symétries de l'état juste après l'application d'une règle d'interaction à l'état juste avant d'avoir appliqué celle-ci. Comme les configurations de protéines qui ne satisfont les conditions contextuelles sont réduites à des singletons, cela revient à dire que l'application d'une règle d'interaction à deux configurations distinctes de protéines doit produire deux configurations symétriques de la protéines, dès lors que cette règle instaure la condition contextuelle.

Exemple 6.5.5 *Dans l'exemple des figures 6.36 et 6.37, les états des sites d et g dans les occurrences des configurations de la protéine A dans lesquelles le site h est non phosphorylé, ne sont pas indépendants. Si le site h pouvait se faire déphosphoryler sans condition, cette correlation induirait une corrélation entre les états des sites d et g dans les occurrences des configurations de la protéine A dans lesquelles le site h est phosphorylé. Il n'y aurait alors pas de bisimulation en arrière.*

Étudier, inférer et exploiter les symétries contextuelles est une voie pleine de promesses. Il est possible de spécifier quels contextes distingués en dépliant la carte de contacts, comme cela l'a déjà été fait dans la section 4.6 pour la réduction de la sémantique différentielle. Cela reste néanmoins un paramètre que le concepteur du modèle doit fournir à la main. Il est sans doute possible de rendre cette paramétrisation automatique en construisant le dépliage de la carte de contacts à partir des motifs qui apparaissent dans les règles d'interaction. Des heuristiques sont à trouver pour trouver un compromis intéressant entre précision de l'analyse (et donc capacité de réduction) et temps de calcul de celle-ci.

Chapitre 7

Conclusion

Après un bref passage en revue de l'écosystème Kappa et de l'utilisation de l'interprétation abstraite pour extraire les propriétés des réseaux d'interactions biomoléculaires, le langage Kappa a été présenté plus en détail, ainsi qu'une analyse statique pour détecter parmi un ensemble de motifs d'intérêt lesquels peuvent potentiellement apparaître dans des configurations d'espèces biochimiques dans une trace d'exécution d'un modèle et des méthodes de réduction de modèles pour réduire la combinatoire de ces modèles.

Du point de vue de l'utilisateur, l'analyse statique permet de trouver – ou de retrouver – des propriétés structurelles sur les différentes configurations des occurrences des protéines au sein des complexes biochimiques : elle détecte quelles sont les relations entre l'état des sites des occurrences d'une protéine (Est-ce que tel site peut être lié sans que tel autre ne le soit ? Est-ce que ce site peut être lié sans être phosphorylé ?) ; elle permet de vérifier si deux occurrences de protéines liées entre-elles sont, oui ou non, nécessairement localisées au même endroit au sein d'une hiérarchie statique de compartiments ; elle analyse si une occurrence de protéines peut être doublement liée à une autre ou si elle peut être liée à deux occurrences différentes de protéines. En plus, de permettre la détection de règles mortes, qui ne pourront jamais être appliquées dans le modèle, le résultat est présenté graphiquement sous la forme de lemmes de raffinement, ce qui le rend compréhensible et facilement utilisable pour des analyses ultérieures. Il est ensuite possible de se concentrer sur le comportement des occurrences d'une protéine en particulier et d'obtenir un système de transitions pour décrire leurs changements potentiels de configuration. Cette analyse passe à l'échelle de grands modèles. Cependant, pour ceux-ci, le temps de calcul reste trop important pour permettre une analyse interactive et sans latence pendant l'écriture même des modèles. Une formulation du calcul du plus petit point fixe abstrait sous forme de résolution de clauses de Horn pourrait donner lieu à une analyse incrémentale. Celle-ci permettrait de mettre à jour très rapidement le résultat de l'analyse lorsque des règles sont retirées ou ajoutées à un modèle.

Par ailleurs, une collaboration étroite avec les modélisateurs est toujours nécessaire pour identifier des nouvelles familles de propriétés d'intérêt. Un autre axe de recherche est l'intégration de l'analyse statique dans des cycles de modélisations automatiques. En effet, les méthodes de fouille de la littérature basées sur l'intelligence artificielle et le traitement automatique des langages naturels pourront bénéficier de l'analyse statique d'une part pour évaluer le bien fondé d'une étape de raffinement de modèle et d'autre part pour orienter les méthodes automatiques dans leur recherche de nouvelles règles. En ce qui concerne la modélisation en Kappa, il est important de considérer non pas un réseau d'interactions biomoléculaires dans son individualité, mais une famille de réseaux d'interactions pouvant représenter un système dans différents contextes cellulaires et ses évolutions potentielles. Les travaux sur la plate-forme de modélisation Kami vont dans ce sens [92, 94]. Il est aussi important de proposer des méthodes pour assister le modélisateur dans la construction de modèles, afin d'agglomérer des informations partielles sur les interactions biomoléculaires en les raffinant progressivement. Une approche inspirée des approches déductives, qui assimile le processus de modélisation à une recherche de preuves assistée par ordinateur, est très prometteuse [98, 97]. Dans ce contexte, une analyse statique le plus tôt possible dans la chaîne de modélisation doit être développée pour aider au mieux le modélisateur dans sa tâche.

Améliorer l'interactivité des outils [14, 19] et un travail sur le rendu visuel des propriétés [82] sont des enjeux cruciaux pour créer des outils utilisables pour des modélisateurs non experts en langage formel. Il faut intéresser un spectre plus large d'utilisateurs. D'une part, c'est une source inépuisable de défis scientifiques. D'autre part, c'est nécessaire pour construire un nombre satisfaisant de modèles.

En ce qui concerne la réduction de modèles, deux approches ont été présentées.

1. La première se base sur l'étude des flots d'information. Il s'agit de détecter les états de quels sites influ-

encent l'évolution de tels autres dans les configurations des espèces biochimiques d'un modèle. De cette analyse, peuvent être déduits des découpages des configurations d'espèces biochimiques soit dont les états ne sont pas corrélés, soit dont la corrélation n'est pas importante pour suivre le comportement du modèle. Ceux-ci induisent des changements de variables qui réduisent la dimension des systèmes engendrés par les modèles. Cette approche se décline pour la sémantique différentielle et pour la sémantique stochastique. Cependant, alors qu'elle donne des résultats très prometteurs dans le cadre différentiel, elle ne réduit quasiment pas les modèles dans le cadre stochastique. Ceci est du à deux phénomènes. En stochastique, des pas de calculs peuvent agir de manière conjointe sur deux morceaux de configurations d'espèces biochimiques, ce qui impose d'en connaître les corrélations. Ceci ne pose aucun problème dans le cadre différentiel, où les actions s'appliquent, pour chacun des deux morceaux, sur une proportion de ceux-ci, indépendamment de leur distribution jointe. Par ailleurs, des termes de différents ordres de grandeur apparaissent dans la sémantique stochastique. Certains sont à l'origine de petits flots d'information qui établissent des corrélations entre les états de différents morceaux de configurations des espèces biochimiques. Ces termes disparaissent dans la sémantique différentielle à la limite thermodynamique.

2. La seconde est basée sur l'étude des symétries entre les sites d'interaction des espèces biochimiques. Un cadre algébrique permet de définir quand plusieurs sites ont exactement le même comportement dans un modèle. Il en découle une relation de bisimulation avant-arrière qui permet de réduire les sémantiques différentielles et stochastiques des modèles en conséquence, essentiellement en imposant l'ordre selon lequel ces sites indiscernables sont activés. L'aspect bisimulation en avant assure que les états symétriques ont le même comportement vis à vis des classes de symétries des états. Alors que l'aspect bisimulation en arrière assure l'existence d'invariants quantitatifs quand les états initiaux ou les distributions initiales d'états satisfont eux aussi les symétries en question. Cette approche peut être étendue à des sous-groupes de symétries pour traiter des symétries plus exotiques comme celles rencontrées en stéréochimie ou comme les équivalences entre sites qui ne sont valables que sous certaines conditions sur l'état des autres sites. Dans ce cas, les symétries d'un modèle n'induisent pas forcément une bisimulation en avant, une bisimulation en arrière, ou une bisimulation avant-arrière. Il faut alors étudier si les transformations appliquées à la partie commune entre l'état juste avant d'appliquer chaque pas de calcul et celui juste après avoir appliqué celui-ci peuvent s'étendre à des transformations qui s'appliquent à l'état juste avant l'application de ce pas de calcul (pour la bisimulation arrière) ou juste après (pour la bisimulation en avant).

Tout comme l'analyse des symétries, l'analyse de flot peut être affinée en la rendant plus ou moins sensible au contexte, c'est à dire aux états des autres sites dans les occurrences des configurations des espèces biochimiques. La spécification de l'ensemble des contextes à distinguer reste alors à la charge de l'utilisateur. Il sera important de proposer des heuristiques pour automatiser ce paramètre, afin de proposer une large gamme de compromis entre le temps de l'analyse et sa capacité de réduction.

Les modèles sont de plus en plus grands, que ce soit en nombre de configurations d'espèces biochimiques différentes ou en nombre d'instances des complexes biochimiques. Évaluer leurs comportements est primordial, mais difficile. Les méthodes exactes de réduction de modèles sont utiles, mais limitées, pour ce type de modèles. Il est important de développer des méthodes numériques approchées pour les sémantiques différentielles et stochastiques des modèles qui permettront de trouver un encadrement garanti de l'évolution du nombre d'instances ou de la concentration, selon le choix de la sémantique, de motifs d'intérêt au cours du temps, sous la forme de paires de fonctions, elles-mêmes définies comme la solution d'un système différentiel ou comme les trajectoires d'un système stochastique. Des travaux préliminaires ont permis d'intégrer dans un cadre formel des méthodes de troncation de développement formel [123] ou des méthodes inspirées de la physique comme la tropicalisation [7], tout en fournissant des bornes évoluant au cours de l'exécution des modèles sur les erreurs numériques accumulées. Il devrait également être possible de définir une version quantitative de l'analyse de flot d'information entre sites des protéines, afin de négliger les petits flots d'information, au prix d'une perte de précision dans les modèles réduits. Un cadre formel pour l'exécution numériquement approchée des modèles permettra d'interfacer les sémantiques différentielles et stochastiques de Kappa pour concevoir une sémantique hybride, plus adaptée à la description des interactions entre des complexes biochimiques géants rares et des petits complexes présents en très grand nombre.

Références bibliographiques

[1] Wassim Abou-Jaoudé, Jérôme Feret et Denis Thieffry : Derivation of qualitative dynamical models from biochemical networks. *In* Olivier F. Roux et Jérémie Bourdon, éditeurs : *Computational Methods in Systems Biology - 13th International Conference, CMSB 2015, Nantes, France, September 16-18, 2015, Proceedings*, volume 9308 de *Lecture Notes in Computer Science*, pages 195–207. Springer, 2015.

[2] Wassim Abou-Jaoudé, Denis Thieffry et Jérôme Feret : Formal derivation of qualitative dynamical models from biochemical networks. *Biosystems*, 149:70–112, 2016.

[3] Emilie Allart, Joachim Niehren et Cristian Versari : Computing difference abstractions of metabolic networks under kinetic constraints. *In* Luca Bortolussi et Guido Sanguinetti, éditeurs : *Computational Methods in Systems Biology - 17th International Conference, CMSB 2019, Trieste, Italy, September 18-20, 2019, Proceedings*, volume 11773 de *Lecture Notes in Computer Science*, pages 266–285. Springer, 2019.

[4] Jakob L. Andersen, Christoph Flamm, Daniel Merkle et Peter F. Stadler : A software package for chemically inspired graph transformation. *In* Rachid Echahed et Mark Minas, éditeurs : *Graph Transformation - 9th International Conference, ICGT 2016, in Memory of Hartmut Ehrig, Held as Part of STAF 2016, Vienna, Austria, July 5-6, 2016, Proceedings*, volume 9761 de *Lecture Notes in Computer Science*, pages 73–88. Springer, 2016.

[5] Oana Andrei et Hélène Kirchner : A rewriting calculus for multigraphs with ports. *Electr. Notes Theor. Comput. Sci.*, 219:67–82, 2008.

[6] Nicolas Behr et Jean Krivine : Compositionality of rewriting rules with conditions. *CoRR*, abs/1904.09322, 2019.

[7] Andreea Beica, Jérôme Feret et Tatjana Petrov : Tropical abstraction of biochemical reaction networks with guarantees. *Electr. Notes Theor. Comput. Sci.*, 350:3–32, 2020.

[8] Bruno Blanchet, Patrick Cousot, Radhia Cousot, Jérôme Feret, Laurent Mauborgne, Antoine Miné, David Monniaux et Xavier Rival : A static analyzer for large safety-critical software. *In* Ron Cytron et Rajiv Gupta, éditeurs : *Proceedings of the ACM SIGPLAN 2003 Conference on Programming Language Design and Implementation 2003, San Diego, California, USA, June 9-11, 2003*, pages 196–207, 2003.

[9] Michael L. Blinov, James R. Faeder, Byron Goldstein et William S. Hlavacek : Bionetgen: software for rule-based modeling of signal transduction based on the interactions of molecular domains. *Bioinformatics*, 20(17), 2004.

[10] Michael L. Blinov, James R. Faeder, Byron Goldstein et William S. Hlavacek : A network model of early events in epidermal growth factor receptor signaling that accounts for combinatorial complexity. *BioSystems*, 83:136–151, 2006.

[11] Chiara Bodei, Linda Brodo, Roberta Gori, Diana Hermith et Francesca Levi : A global occurrence counting analysis for brane calculi. *In* Moreno Falaschi, éditeur : *Logic-Based Program Synthesis and Transformation - 25th International Symposium, LOPSTR 2015, Siena, Italy, July 13-15, 2015. Revised Selected Papers*, volume 9527 de *Lecture Notes in Computer Science*, pages 179–200. Springer, 2015.

[12] Chiara Bodei, Pierpaolo Degano, Flemming Nielson et Hanne Riis Nielson : Control flow analysis for the pi-calculus. *In* Davide Sangiorgi et Robert de Simone, éditeurs : *CONCUR '98: Concurrency Theory, 9th International Conference, Nice, France, September 8-11, 1998, Proceedings*, volume 1466 de *Lecture Notes in Computer Science*, pages 84–98. Springer, 1998.

[13] Nikolay M. Borisov, Nick I. Markevich, Boris N. Kholodenko et Ernst Dieter Gilles : Signaling through receptors and scaffolds: Independent interactions reduce combinatorial complexity. *Biophysical Journal*, 89, 2005.

[14] Pierre Boutillier : The kappa simulator made interactive. *In* Luca Bortolussi et Guido Sanguinetti, éditeurs : *Computational Methods in Systems Biology - 17th International Conference, CMSB 2019, Trieste, Italy, September 18-20, 2019, Proceedings*, volume 11773 de *Lecture Notes in Computer Science*, pages 296–301. Springer, 2019.

[15] Pierre Boutillier, Ferdinanda Camporesi, Jean Coquet, Jérôme Feret, Kim Quyên Lý, Nathalie Théret et Pierre Vignet : Kasa: A static analyzer for kappa. *In* Milan Ceska et David Safránek, éditeurs : *Computational Methods in Systems Biology - 16th International Conference, CMSB 2018, Brno, Czech Republic, September 12-14, 2018, Proceedings*, volume 11095 de *Lecture Notes in Computer Science*, pages 285–291. Springer, 2018.

[16] Pierre Boutillier, Ioana Cristescu et Jérôme Feret : Counters in kappa: Semantics, simulation, and static analysis. *In* Luís Caires, éditeur : *Programming Languages and Systems - 28th European Symposium on Programming, ESOP 2019, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2019, Prague, Czech Republic, April 6-11, 2019, Proceedings*, volume 11423 de *Lecture Notes in Computer Science*, pages 176–204. Springer, 2019.

[17] Pierre Boutillier, Thomas Ehrhard et Jean Krivine : Incremental update for graph rewriting. *In* Hongseok Yang, éditeur : *Programming Languages and Systems - 26th European Symposium on Programming, ESOP 2017, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2017, Uppsala, Sweden, April 22-29, 2017, Proceedings*, volume 10201 de *Lecture Notes in Computer Science*, pages 201–228. Springer, 2017.

[18] Pierre Boutillier, Aurélie Faure de Pebeyre et Jérôme Feret : Proving the absence of unbounded polymers in rule-based models. *In Nine International Workshop on Static Analysis and Systems Biology (SASB'18)*, volume 350 de *ENTCS*, pages 33–56. elsevier, 2020.

[19] Pierre Boutillier, Mutaamba Maasha, Xing Li, Héctor F. Medina-Abarca, Jean Krivine, Jérôme Feret, Ioana Cristescu, Angus G. Forbes et Walter Fontana : The kappa platform for rule-based modeling. *Bioinformatics*, 34(13):i583–i592, 2018.

[20] Frances A. Brightman et David A. Fell : Differential feedback regulation of the mapk cascade underlies the quantitative differences in egf and ngf signalling in pc12 cells. *FEBS Letters*, 482(3):169–174, 2000.

[21] Peter Buchholz : Exact and ordinary lumpability in finite markov chains. *Journal of Applied Probability*, 31(1):59–75, 1994.

[22] Peter Buchholz : Bisimulation relations for weighted automata. *TCS*, 393(1-3):109–123, 2008.

[23] Ferdinanda Camporesi : *Formal and exact reduction for differential models of signalling pathways in rule-based languages.* Thèse de doctorat, Paris Sciences et Lettres Research University, January 2017.

[24] Ferdinanda Camporesi et Jérôme Feret : Formal reduction for rule-based models. *In* Michael W. Mislove et Joël Ouaknine, éditeurs : *Twenty-seventh Conference on the Mathematical Foundations of Programming Semantics, MFPS 2011, Pittsburgh, PA, USA, May 25-28, 2011*, volume 276 de *Electronic Notes in Theoretical Computer Science*, pages 29–59. Elsevier, 2011.

[25] Ferdinanda Camporesi et Jérôme Feret : Using alternated sums to express the occurrence number of extended patterns in site-graphs. *In* Jean Yang et John A. Bachman, éditeurs : *SASB 2017 - The Eighth International Workshop on Static Analysis for Systems Biology*, Static Analysis and Systems Biology, page 18, New York, United States, août 2017. Elsevier. To appear.

[26] Ferdinanda Camporesi, Jérôme Feret et Jonathan Hayman : Context-sensitive flow analyses: A hierarchy of model reductions. *In* Ashutosh Gupta et Thomas A. Henzinger, éditeurs : *Computational Methods in Systems Biology - 11th International Conference, CMSB 2013, Klosterneuburg, Austria, September 22-24, 2013. Proceedings*, volume 8130 de *Lecture Notes in Computer Science*, pages 220–233. Springer, 2013.

[27] Ferdinanda Camporesi, Jérôme Feret, Heinz Koeppl et Tatjana Petrov : Combining model reductions. *In MFPSXXVI: Postproceedings of the 26th Conference on the Mathematical Foundations of Programming Semantics*, volume 265 de *Electronic Notes in Theoretical Computer Science*, pages 73–96. Elsevier Science Publishers, 2010.

[28] Ferdinanda Camporesi, Jérôme Feret et Kim Quyên Lý : KaDe: a tool to compile kappa rules into (reduced) ode models: Supplementary information.

[29] Ferdinanda Camporesi, Jérôme Feret et Kim Quyên Lý : Kade: A tool to compile kappa rules into (reduced) ODE models. *In* Jérôme Feret et Heinz Koeppl, éditeurs : *Computational Methods in Systems Biology - 15th International Conference, CMSB 2017, Darmstadt, Germany, September 27-29, 2017, Proceedings*, volume 10545 de *Lecture Notes in Computer Science*, pages 291–299. Springer, 2017.

[30] Luca Cardelli : Brane calculi. *In* Vincent Danos et Vincent Schächter, éditeurs : *Computational Methods in Systems Biology, International Conference, CMSB 2004, Paris, France, May 26-28, 2004, Revised Selected Papers*, volume 3082 de *Lecture Notes in Computer Science*, pages 257–278. Springer, 2004.

[31] Luca Cardelli et Andrew D. Gordon : Mobile ambients. *In* Maurice Nivat, éditeur : *Foundations of Software Science and Computation Structure, First International Conference, FoSSaCS'98, Held as Part of the European Joint Conferences on the Theory and Practice of Software, ETAPS'98, Lisbon, Portugal, March 28 - April 4, 1998, Proceedings*, volume 1378 de *Lecture Notes in Computer Science*, pages 140–155. Springer, 1998.

[32] Luca Cardelli et Andrew D. Gordon : Mobile ambients. *Theor. Comput. Sci.*, 240(1):177–213, 2000.

[33] Luca Cardelli, Giuseppe Squillace, Mirco Tribastone, Max Tschaikowski et Andrea Vandin : Formal lumping of polynomial differential equations through approximate equivalences. *J. Log. Algebraic Methods Program.*, 134:100876, 2023.

[34] Luca Cardelli, Mirco Tribastone, Max Tschaikowski et Andrea Vandin : Forward and backward bisimulations for chemical reaction networks. *In* Luca Aceto et David de Frutos-Escrig, éditeurs : *Proc. CONCUR'15*, volume 42 de *LIPIcs*, pages 226–239. Schloss Dagstuhl, 2015.

[35] Luca Cardelli, Mirco Tribastone, Max Tschaikowski et Andrea Vandin : Efficient syntax-driven lumping of differential equations. *In* Marsha Chechik et Jean-François Raskin, éditeurs : *Proc. TACAS'16*, volume 9636 de *LNCS*, pages 93–111. Springer, 2016.

[36] Luca Cardelli, Mirco Tribastone, Max Tschaikowski et Andrea Vandin : Symbolic computation of differential equivalences. *Theor. Comput. Sci.*, 777:132–154, 2019.

[37] Federica Ciocchetta et Jane Hillston : Bio-PEPA: A framework for the modelling and analysis of biological systems. *Theoretical Computer Science*, 410(33 – 34):3065 – 3084, 2009. Concurrent Systems Biology: To Nadia Busi (1968–2007).

[38] Paul R. Cohen : DARPA's big mechanism program. *Physical Biology*, 12(4):045008, jul 2015.

[39] Holger Conzelmann, Dirk Fey et Ernst D. Gilles : Exact model reduction of combinatorial reaction networks. *BMC Systems Biology*, 2:78, 2008.

[40] Holger Conzelmann, Julio Saez-Rodriguez, Thomas Sauter, Boris N. Kholodenko et Ernst D. Gilles : A domain-oriented approach to the reduction of combinatorial complexity in signal transduction networks. *BMC Bioinformatics*, 7, 2006.

[41] Byron Cook, Jasmin Fisher, Elzbieta Krepska et Nir Piterman : Proving stabilization of biological systems. *In* Ranjit Jhala et David A. Schmidt, éditeurs : *Verification, Model Checking, and Abstract Interpretation - 12th International Conference, VMCAI 2011, Austin, TX, USA, January 23-25, 2011. Proceedings*, volume 6538, pages 134–149. Springer, 2011.

[42] Andrea Corradini, Tobias Heindel, Frank Hermann et Barbara König : Sesqui-pushout rewriting. *In* Andrea Corradini, Hartmut Ehrig, Ugo Montanari, Leila Ribeiro et Grzegorz Rozenberg, éditeurs : *Graph Transformations, Third International Conference, ICGT 2006, Natal, Rio Grande do Norte, Brazil, September 17-23, 2006, Proceedings*, volume 4178 de *Lecture Notes in Computer Science*, pages 30–45. Springer, 2006.

[43] Andrea Corradini, Ugo Montanari, Francesca Rossi, Hartmut Ehrig, Reiko Heckel et Michael Löwe : Algebraic approaches to graph transformation - part I: basic concepts and double pushout approach. *In* Grzegorz Rozenberg, éditeur : *Handbook of Graph Grammars and Computing by Graph Transformations, Volume 1: Foundations*, pages 163–246. World Scientific, 1997.

[44] Patrick Cousot : The calculational design of a generic abstract interpreter. *In* M. Broy et R. Steinbrüggen, éditeurs : *Calculational System Design*, pages 1–88. NATO ASI Series F. IOS Press, Amsterdam, 1999.

[45] Patrick Cousot : Constructive design of a hierarchy of semantics of a transition system by abstract interpretation. *Theoretical Computer Science*, 277(1–2):47–103, 2002.

[46] Patrick Cousot et Radhia Cousot : Abstract interpretation: A unified lattice model for static analysis of programs by construction or approximation of fixpoints. *In* Robert M. Graham, Michael A. Harrison et Ravi Sethi, éditeurs : *Conference Record of the Fourth ACM Symposium on Principles of Programming Languages, Los Angeles, California, USA, January 1977*, pages 238–252. ACM, 1977.

[47] Patrick Cousot et Radhia Cousot : Systematic design of program analysis frameworks. *In* Alfred V. Aho, Stephen N. Zilles et Barry K. Rosen, éditeurs : *Conference Record of the Sixth Annual ACM Symposium on Principles of Programming Languages, San Antonio, Texas, USA, January 1979*, pages 269–282. ACM Press, 1979.

[48] Troels Christoffer Damgaard, Espen Højsgaard et Jean Krivine : Formal cellular machinery. *Electr. Notes Theor. Comput. Sci.*, 284:55–74, 2012.

[49] Werner Damm et David Harel : LSCs: Breathing life into message sequence charts. *Formal Methods in System Design*, 19(1):45–80, 2001.

[50] Vincent Danos, Jérôme Feret, Walter Fontana, Russell Harmer, Jonathan Hayman, Jean Krivine, Christopher D. Thompson-Walsh et Glynn Winskel : Graphs, rewriting and pathway reconstruction for rule-based models. *In* Deepak D'Souza, Telikepalli Kavitha et Jaikumar Radhakrishnan, éditeurs : *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2012, December 15-17, 2012, Hyderabad, India*, volume 18 de *LIPIcs*, pages 276–288. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2012.

[51] Vincent Danos, Jérôme Feret, Walter Fontana, Russell Harmer et Jean Krivine : Rule-based modelling of cellular signalling. *In* Luís Caires et Vasco Thudichum Vasconcelos, éditeurs : *CONCUR 2007 - Concurrency Theory, 18th International Conference, CONCUR 2007, Lisbon, Portugal, September 3-8, 2007, Proceedings*, volume 4703 de *Lecture Notes in Computer Science*, pages 17–41. Springer, 2007.

[52] Vincent Danos, Jérôme Feret, Walter Fontana, Russell Harmer et Jean Krivine : Rule-based modelling, symmetries, refinements. *In* Jasmin Fisher, éditeur : *Formal Methods in Systems Biology, First International Workshop, FMSB 2008, Cambridge, UK, June 4-5, 2008. Proceedings*, volume 5054 de *Lecture Notes in Computer Science*, pages 103–122. Springer, 2008.

[53] Vincent Danos, Jérôme Feret, Walter Fontana, Russell Harmer et Jean Krivine : Abstracting the differential semantics of rule-based models: Exact and automated model reduction. *In Proceedings of the 25th Annual IEEE Symposium on Logic in Computer Science, LICS 2010, 11-14 July 2010, Edinburgh, United Kingdom*, pages 362–381. IEEE Computer Society, 2010.

[54] Vincent Danos, Jérôme Feret, Walter Fontana et Jean Krivine : Scalable simulation of cellular signaling networks. *In* Zhong Shao, éditeur : *Programming Languages and Systems, 5th Asian Symposium, APLAS 2007, Singapore, November 29-December 1, 2007, Proceedings*, volume 4807 de *Lecture Notes in Computer Science*, pages 139–157. Springer, 2007.

[55] Vincent Danos, Jérôme Feret, Walter Fontana et Jean Krivine : Scalable simulation of cellular signaling networks, invited paper. *In APLAS'07: Proceedings of the Fifth Asian Symposium on Programming Systems*, volume 4807 de *Lecture Notes in Computer Science*, pages 139–157. Springer, Berlin, Germany, 2007.

[56] Vincent Danos, Jérôme Feret, Walter Fontana et Jean Krivine : Abstract interpretation of cellular signalling networks. *In* Francesco Logozzo, Doron A. Peled et Lenore D. Zuck, éditeurs : *Verification, Model Checking, and Abstract Interpretation, 9th International Conference, VMCAI 2008, San Francisco, USA, January 7-9, 2008, Proceedings*, volume 4905 de *Lecture Notes in Computer Science*, pages 83–97. Springer, 2008.

[57] Vincent Danos, Ricardo Honorato-Zimmer, Se Jaramillo-Riveri et Sandro Stucki : Rigid geometric constraints for Kappa models. *In SASB'12: PostProceedings of the 3rd International Workshop on Static Analysis and Systems Biology*, volume 313 de *ENTCS*, pages 23–46. Elsevier, 2015.

[58] Vincent Danos et Cosimo Laneve : Graphs for core molecular biology. *In* Corrado Priami, éditeur : *Computational Methods in Systems Biology, First International Workshop, CMSB 2003, Roverto, Italy, February 24-26, 2003, Proceedings*, volume 2602 de *Lecture Notes in Computer Science*, pages 34–46. Springer, 2003.

[59] Vincent Danos et Cosimo Laneve : Formal molecular biology. *Theoretical Computer Science*, 325(1):69 – 110, 2004. Computational Systems Biology.

[60] Vincent Danos et Sylvain Pradalier : Projective brane calculus. *In* Vincent Danos et Vincent Schächter, éditeurs : *Computational Methods in Systems Biology, International Conference, CMSB 2004, Paris, France, May 26-28, 2004, Revised Selected Papers*, volume 3082 de *Lecture Notes in Computer Science*, pages 134–148. Springer, 2004.

[61] Tadeas Ded, David Safránek, Matej Troják, Matej Klement, Jakub Salagovic et Lubos Brim : Formal biochemical space with semantics in kappa and BNGL. *Electr. Notes Theor. Comput. Sci.*, 326:27–49, 2016.

[62] Lorenzo Dematté, Corrado Priami et Alessandro Romanel : The blenx language: A tutorial. *In* Marco Bernardo, Pierpaolo Degano et Gianluigi Zavattaro, éditeurs : *Formal Methods for Computational Systems Biology: 8th International School on Formal Methods for the Design of Computer, Communication, and Software Systems, SFM 2008 Bertinoro, Italy, June 2-7, 2008 Advanced Lectures*, pages 313–365, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.

[63] Josée Desharnais, Vineet Gupta, Radha Jagadeesan et Prakash Panangaden : Metrics for labeled markov systems. *In* Jos C. M. Baeten et Sjouke Mauw, éditeurs : *CONCUR '99: Concurrency Theory, 10th International Conference, Eindhoven, The Netherlands, August 24-27, 1999, Proceedings*, volume 1664 de *Lecture Notes in Computer Science*, pages 258–273. Springer, 1999.

[64] Jacob L. Doob : Markoff chains–denumerable case. *Transactions of the American Mathematical Society*, 58(3):455–473, 1945.

[65] James R. Faeder, Michael L. Blinov, Byron Goldstein et William S. Hlavacek : Rule-based modeling of biochemical networks. *Complexity*, 10(4):22–41, 2005.

[66] Manuel Fähndrich et Francesco Logozzo : Static contract checking with abstract interpretation. *In* Bernhard Beckert et Claude Marché, éditeurs : *Formal Verification of Object-Oriented Software - International Conference, FoVeOOS 2010, Paris, France, June 28-30, 2010, Revised Selected Papers*, volume 6528 de *LNCS*, pages 10–30. Springer, 2010.

[67] Martin Feinberg : Lectures on chemical reaction networks, 1979. Notes of lectures given at the Mathematics Research Centre, University of Wisconsin, in 1979.

[68] Willy Feller : On the integro-differential equations of purely discontinuous markoff processes. *Transactions of the American Mathematical Society*, 48(3):488–515, 1940.

[69] Jérôme Feret : Confidentiality analysis of mobile systems. *In* Jens Palsberg, éditeur : *Static Analysis, 7th International Symposium, SAS 2000, Santa Barbara, CA, USA, June 29 - July 1, 2000, Proceedings*, volume 1824 de *Lecture Notes in Computer Science*, pages 135–154. Springer, 2000.

[70] Jérôme Feret : Occurrence counting analysis for the pi-calculus. *Electr. Notes Theor. Comput. Sci.*, 39(2):1–18, 2001.

[71] Jérôme Feret : Reachability analysis of biological signalling pathways by abstract interpretation. *In Proc. ICCMSE'07*, volume 963 de *AIP*, 2007.

[72] Jérôme Feret : Fragments-based model reduction: some case studies. *In* Jean Krivine et Angelo Troina, éditeurs : *Préproceedings of the First International Workshop on Interactions between Computer Science and Biology, CS2Bio '2010*, volume 268 de *Electonic Notes in Theoretical Computer Science*, pages 77–96, Amsterdam, Netherlands, 10 June 2010. Elsevier Science Publishers.

[73] Jérôme Feret : An algebraic approach for inferring and using symmetries in rule-based models. *In* Loïc Paulevé et Heinz Koeppl, éditeurs : *5th International Workshop on Static Analysis and Systems Biology (SASB 2014)*, volume 316 de *ENTCS*, pages 45–65. Elsevier, 2014.

[74] Jérôme Feret, Vincent Danos, Jean Krivine, Russ Harmer et Walter Fontana : Internal coarse-graining of molecular systems. *PNAS*, 2009.

[75] Jérôme Feret, Thomas A. Henzinger, Heinz Koeppl et Tatjana Petrov : Lumpability abstractions of rule-based systems. *Theor. Comput. Sci.*, 431:137–164, 2012.

[76] Jérôme Feret, Heinz Koeppl et Tatjana Petrov : Stochastic fragments: A framework for the exact reduction of the stochastic semantics of rule-based models. *Int. J. Software and Informatics*, 7(4):527–604, 2013.

[77] Jérôme Feret et Kim Quyên Lý : Local traces: An over-approximation of the behaviour of the proteins in rule-based models. *In* Ezio Bartocci, Pietro Liò et Nicola Paoletti, éditeurs : *Computational Methods in Systems Biology - 14th International Conference, CMSB 2016, Cambridge, UK, September 21-23, 2016, Proceedings*, volume 9859 de *Lecture Notes in Computer Science*, pages 116–131. Springer, 2016.

[78] Jérôme Feret et Kim Quyên Lý : Local traces: An over-approximation of the behavior of the proteins in rule-based models. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 15(4):1124–1137, July-Aug. 2018.

[79] Jérôme Feret et Kim Quyên Lý : Reachability analysis via orthogonal sets of patterns. *Electr. Notes Theor. Comput. Sci.*, 335:27–48, 2018.

[80] Norm Ferns, Prakash Panangaden et Doina Precup : Bisimulation metrics for continuous markov decision processes. *SIAM J. Comput.*, 40(6):1662–1714, 2011.

[81] Maxime Folschette, Loïc Paulevé, Morgan Magnin et Olivier F. Roux : Under-approximation of reachability in multivalued asynchronous networks. *Electr. Notes Theor. Comput. Sci.*, 299:33–51, 2013.

[82] Angus Graeme Forbes, Andrew Burks, Kristine Lee, Xing Li, Pierre Boutillier, Jean Krivine et Walter Fontana : Dynamic influence networks for rule-based models. *IEEE Trans. Vis. Comput. Graph.*, 24(1):184–194, 2018.

[83] Qian Gao, Fei Liu, David Gilbert, Monika Heiner et David Tree : A multiscale approach to modelling planar cell polarity in drosophila wing using hierarchically coloured petri nets. *In Proceedings of the 9th International Conference on Computational Methods in Systems Biology*, CMSB '11, pages 209–218, New York, NY, USA, 2011. ACM.

[84] Steven Gay, François Fages, Thierry Martinez, Sylvain Soliman et Christine Solnon : On the subgraph epimorphism problem. *Discrete Applied Mathematics*, 162:214–228, 2014.

[85] Colin S. Gillespie : Moment-closure approximations for mass-action models. *IET systems biology*, 3(1):52–58, 2009.

[86] Daniel T. Gillespie : Exact stochastic simulation of coupled chemical reactions. *The Journal of Physical Chemistry*, 81(25):2340–2361, 1977.

[87] Roberta Gori et Francesca Levi : An analysis for proving temporal properties of biological systems. *In* Naoki Kobayashi, éditeur : *Programming Languages and Systems, 4th Asian Symposium, APLAS 2006, Sydney, Australia, November 8-10, 2006, Proceedings*, volume 4279 de *Lecture Notes in Computer Science*, pages 234–252. Springer, 2006.

[88] Radu Grosu, Grégory Batt, Flavio H. Fenton, James Glimm, Colas Le Guernic, Scott A. Smolka et Ezio Bartocci : From cardiac cells to genetic regulatory networks. *In* Ganesh Gopalakrishnan et Shaz Qadeer, éditeurs : *Computer Aided Verification - 23rd International Conference, CAV 2011, Snowbird, UT, USA, July 14-20, 2011. Proceedings*, volume 6806 de *Lecture Notes in Computer Science*, pages 396–411. Springer, 2011.

[89] Benjamin Gyori, John Bachman, Kartik Subramanian, Jeremy Muhlich, Lucian Galescu et Peter Sorger : From word models to executable models of signaling networks using automated assembly. *Molecular Systems Biology*, 13, 2017.

[90] Joseph Y. Halpern et Judea Pearl : Causes and explanations: A structural-model approach — part 1: Causes. *CoRR*, abs/1301.2275, 2013.

[91] Russ Harmer : Rule-based modelling and tunable resolution. *In DCM'09: Proceedings Fifth Workshop on Developments in Computational Models–Computational Models From Nature*, volume 9 de *EPTCS*, pages 65–72, 2009.

[92] Russ Harmer, Yves-Stan Le Cornec, Sébastien Légaré et Eugenia Oshurko : Bio-curation for cellular signalling: The KAMI project. *IEEE/ACM Trans. Comput. Biology Bioinform.*, 16(5):1562–1573, 2019.

[93] Russ Harmer, Vincent Danos, Jérôme Feret, Jean Krivine et Walter Fontana : Intrinsic information carriers in combinatorial dynamical systems. *Chaos*, 20, September 2010.

[94] Russ Harmer et Eugenia Oshurko : Kamistudio: An environment for biocuration of cellular signalling knowledge. *In* Luca Bortolussi et Guido Sanguinetti, éditeurs : *Computational Methods in Systems Biology - 17th International Conference, CMSB 2019, Trieste, Italy, September 18-20, 2019, Proceedings*, volume 11773 de *Lecture Notes in Computer Science*, pages 322–328. Springer, 2019.

[95] Monika Heiner et Ina Koch : Petri net based model validation in systems biology. *In* Jordi Cortadella et Wolfgang Reisig, éditeurs : *Applications and Theory of Petri Nets 2004: 25th International Conference, ICATPN 2004, Bologna, Italy, June 21–25, 2004. Proceedings*, pages 216–237. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.

[96] Tobias Helms, Tom Warnke, Carsten Maus et Adelinde M. Uhrmacher : Semantics and efficient simulation algorithms of an expressive multilevel modeling language. *ACM Trans. Model. Comput. Simul.*, 27(2):8:1–8:25, 2017.

[97] Adrien Husson : *Logical foundations of a modelling assistant for molecular biology.* Thèse de doctorat, Université de Paris, France, 2019.

[98] Adrien Husson et Jean Krivine : A tractable logic for molecular biology. *In* Bart Bogaerts, Esra Erdem, Paul Fodor, Andrea Formisano, Giovambattista Ianni, Daniela Inclezan, Germán Vidal, Alicia Villanueva, Marina De Vos et Fangkai Yang, éditeurs : *Proceedings 35th International Conference on Logic Programming (Technical Communications), ICLP 2019 Technical Communications, Las Cruces, NM, USA, September 20-25, 2019.*, volume 306 de *EPTCS*, pages 101–113, 2019.

[99] Edward L. Ince : *Ordinary Differential Equations.* Dover Publications, Inc., 1956.

[100] Mathias John, Cédric Lhoussaine, Joachim Niehren et Cristian Versari : Biochemical reaction rules with constraints. *In Programming Languages and Systems - 20th European Symposium on Programming, ESOP 2011, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2011, Saarbrücken, Germany, March 26-April 3, 2011. Proceedings*, volume 6602 de *Lecture Notes in Computer Science*, pages 338–357. Springer, 2011.

[101] Mathias John, Mirabelle Nebut et Joachim Niehren : Knockout prediction for reaction networks with partial kinetic information. *In* Roberto Giacobazzi, Josh Berdine et Isabella Mastroeni, éditeurs : *Verification, Model Checking, and Abstract Interpretation, 14th International Conference, VMCAI 2013, Rome, Italy, January 20-22, 2013. Proceedings*, volume 7737 de *Lecture Notes in Computer Science*, pages 355–374. Springer, 2013.

[102] Ozan Kahramanogullari et Luca Cardelli : An intuitive modelling interface for systems biology. *Int. J. Software and Informatics*, 7(4):655–674, 2013.

[103] Hannes Klarner, Alexander Bockmayr et Heike Siebert : Computing maximal and minimal trap spaces of boolean networks. *Natural Computing*, 14(4):535–544, 2015.

[104] Agnes Köhler, Jean Krivine et Jakob Vidmar : A rule-based model of base excision repair. *In* Pedro Mendes, Joseph O. Dada et Kieran Smallbone, éditeurs : *Computational Methods in Systems Biology - 12th International Conference, CMSB 2014, Manchester, UK, November 17-19, 2014, Proceedings*, volume 8859 de *Lecture Notes in Computer Science*, pages 173–195. Springer, 2014.

[105] Juraj Kolcák, David Safránek, Stefan Haar et Loïc Paulevé : Parameter space abstraction and unfolding semantics of discrete regulatory networks. *Theor. Comput. Sci.*, 765:120–144, 2019.

[106] Andrey Kolmogoroff : Über die analytischen methoden in der wahrscheinlichkeitsrechnung. *Mathematische Annalen*, 104:415–458, 1931.

[107] Marta Z. Kwiatkowska, Gethin Norman et David Parker : PRISM 4.0: Verification of probabilistic real-time systems. *In* Ganesh Gopalakrishnan et Shaz Qadeer, éditeurs : *Computer Aided Verification - 23rd International Conference, CAV 2011, Snowbird, UT, USA, July 14-20, 2011. Proceedings*, volume 6806 de *Lecture Notes in Computer Science*, pages 585–591. Springer, 2011.

[108] Jonathan Laurent, Jean Yang et Walter Fontana : Counterfactual resimulation for causal analysis of rule-based models. *In* Jérôme Lang, éditeur : *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden.*, pages 1882–1890. ijcai.org, 2018.

[109] Michael Löwe : Algebraic approach to single-pushout graph transformation. *Theor. Comput. Sci.*, 109(1&2):181–224, 1993.

[110] Antoni W. Mazurkiewicz : Traces, histories, graphs: Instances of a process monoid. *In* Michal Chytil et Václav Koubek, éditeurs : *Mathematical Foundations of Computer Science 1984, Praha, Czechoslovakia, September 3-7, 1984, Proceedings*, volume 176 de *Lecture Notes in Computer Science*, pages 115–133. Springer, 1984.

[111] Donald A. McQuarrie : Stochastic approach to chemical kinetics. *Journal of Applied Probability*, 4(3):pp. 413–478, 1967.

[112] Elaine Murphy, Vincent Danos, Jérôme Feret, Jean Krivine et Russell Harmer : *Elements of Computational Systems Biology*, chapitre Rule Based Modelling and Model Refinement. Wiley Book Series on Bioinformatics. John Wiley & Sons, Inc., 2010.

[113] Hanne Riis Nielson et Flemming Nielson : Shape analysis for mobile ambients. *In* Mark N. Wegman et Thomas W. Reps, éditeurs : *POPL 2000, Proceedings of the 27th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, Boston, Massachusetts, USA, January 19-21, 2000*, pages 142–154. ACM, 2000.

[114] Arnold Nordsieck, Willis E. Lamb et George E. Uhlenbeck : On the theory of cosmic-ray showers i the furry model and the fluctuation problem. *Physica*, 7(4):344–360, 1940.

[115] Loïc Paulevé, Morgan Magnin et Olivier F. Roux : Abstract interpretation of dynamics of biological regulatory networks. *Electr. Notes Theor. Comput. Sci.*, 272:43–56, 2011.

[116] Tatjana Petrov : *Formal reductions of stochastic rule-based models of biochemical systems*. Thèse de doctorat, Eth Zürich, 2013.

[117] Tatjana Petrov, Jérôme Feret et Heinz Koeppl : Reconstructing species-based dynamics from reduced stochastic rule-based models. *In* Oliver Rose et Adelinde M. Uhrmacher, éditeurs : *Winter Simulation Conference, WSC '12, Berlin, Germany, December 9-12, 2012*, pages 225:1–225:15. WSC, 2012.

[118] Brigitte Plateau : On the stochastic structure of parallelism and synchronization models for distributed algorithms. *SIGMETRICS Perform. Eval. Rev.*, 13(2):147–154, août 1985.

[119] Ovidiu Radulescu, Alexander N. Gorban, Andrei Zinovyev et Vincent Noel : Reduction of dynamical biochemical reactions networks in computational biology. *Frontiers in Genetics*, 3:131, 2012.

[120] Ovidiu Radulescu, Sergei Vakulenko et Dima Grigoriev : Model reduction of biochemical reactions networks by tropical analysis methods. *Mathematical Modelling of Natural Phenomena*, 10(3):124–138, 2015.

[121] Aviv Regev, E. M. Panina, William Silverman, Luca Cardelli et E. Y. Shapiro : Bioambients: An abstraction for biological compartments. *TCS*, 325(1):141–167, 2004.

[122] Aviv Regev, William Silverman et Ehud Shapiro : Representation and simulation of biochemical processes using the pi-calculus process algebra. *In* R. B. Altman, A. K. Dunker, L. Hunter et T. E. Klein, éditeurs : *Pacific Symposium on Biocomputing, Volume 6*, pages 459–470, Singapore, 2001.

[123] Ken Chanseau Saint-Germain et Jérôme Feret : Conservative numerical approximations of the differential semantics in biological rule- based models, 2016. Master thesis.

[124] Birgit Schoeberl, Claudia Eichler-Jonsson, Ernst D. Gilles et Gertraud Müller : Computational modeling of the dynamics of the map kinase cascade activated by surface and internalized egf receptors. *Nat Biotechnol*, 20(4):370–375, 2002.

[125] Donald Stewart : Spatial biomodelling, 2010. Master thesis, School of Informatics, University of Edinburgh.

[126] Ryan Suderman et Eric J. Deeds : Machines vs. ensembles: effective mapk signaling through heterogeneous sets of protein complexes. *PLoS Computational Biology*, 9, 2013.

[127] Alfred Tarski : A lattice-theoretical fixpoint theorem and its applications. *Pacific J. Math.*, 5(2), 1955.

Index

RÉSUMÉ

Les sciences du logiciel ont un rôle à jouer pour décrire, organiser, exécuter et analyser les systèmes d'interactions moléculaires tels que les voies de signalisation biologiques. Ceux-ci impliquent une grande diversité d'entités biomoléculaires, alors que leurs dynamiques émergent de compétitions pour des ressources communes, d'interactions à différentes échelles de temps et de concentrations et de boucles de rétroactions non linéaires. Comprendre comment le comportement des populations des protéines émerge des interactions individuelles, le saint Graal de la biologie des systèmes, nécessite des langages dédiés offrant des niveaux d'abstractions adaptés et des outils efficaces.

Dans ce manuscrit, nous décrivons la conception d'outils formels pour Kappa, un langage de réécriture de graphes à sites inspiré de la biochimie. En particulier, nous présentons une analyse statique qui calcule des propriétés sur les entités biologiques qui peuvent se former dans les modèles, et ainsi, améliorer notre confiance en ces derniers. Nous présentons de plus une réduction de modèles, basée sur l'étude du flot d'information entre les différentes régions des entités biologiques et sur des symétries éventuelles. Cette approche s'applique à la fois dans le cadre différentiel et stochastique.

ABSTRACT

Software sciences have a role to play in the description, the organization, the execution, and the analysis of the molecular interaction systems such as biological signaling pathways. These systems involve a huge diversity of bio-molecular entities whereas their dynamics may be driven by races for shared resources, interactions at different time- and concentration-scales, and non-linear feedback loops. Understanding how the behavior of the populations of proteins orchestrates itself from their individual interactions, which is the holy grail on systems biology, requires dedicated languages offering adapted levels of abstraction and efficient analysis tools.

In this manuscript, we describe the design of formal tools for Kappa, a site-graph rewriting language inspired by bio-chemistry. In particular, we introduce a static analysis to compute some properties on the biological entities that may arise in models, so as to increase our confidence in them. We also present a model reduction approach based on a study of the flow of information between the different regions of the biological entities and the potential symmetries. This approach is applied both in the differential and in the stochastic semantics.