



HAL
open science

Cycle de vie des modèles discrets et application à la régulation du métabolisme des cellules cancéreuses du pancréas

Laetitia Gibart

► **To cite this version:**

Laetitia Gibart. Cycle de vie des modèles discrets et application à la régulation du métabolisme des cellules cancéreuses du pancréas. Bio-informatique [q-bio.QM]. Université Côte d'Azur, 2022. Français. NNT : 2022COAZ4074 . tel-03940710

HAL Id: tel-03940710

<https://theses.hal.science/tel-03940710>

Submitted on 16 Jan 2023

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

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

THÈSE DE DOCTORAT

Cycle de vie des modèles discrets et application à la régulation du métabolisme des cellules cancéreuses du pancréas

Laetitia GIBART

Laboratoire d'Informatique, Signaux et Systèmes de Sophia-Antipolis (I3S)
UMR7271 Université Côte d'Azur CNRS

**Présentée en vue de l'obtention
du grade de docteur en Informatique
d'Université Côte d'Azur**

Dirigée par : Jean-Paul COMET
Co-encadrée par : Gilles BERNOT, Hélène
COLLAVIZZA

Soutenue le : 16 Novembre 2022

Devant le jury, composé de :
Morgan MAGNIN, PR, LS2N, Nantes
Sabine PERES, PR, LBBE, Lyon
Gilles BERNOT, PR, Université Côte d'azur,
I3S, Sophia-Antipolis
Hélène COLLAVIZZA, MCF, Université
Côte d'azur, I3S, Sophia-Antipolis
Jean-Paul COMET, PR, Université Côte
d'azur, I3S, Sophia-Antipolis
Fariza TAHI, PR, IBISC, EVRY
Jean-Yves TROSSET, PR, Sup'biotech, Vil-
lejuif
Sophie VASSEUR, DR, CRCM Marseille

**CYCLE DE VIE DES MODÈLES DISCRETS ET APPLICATION À LA
RÉGULATION DU MÉTABOLISME DES CELLULES
CANCÉREUSES DU PANCRÉAS**

*Discrete model life cycle and application to the regulation of
pancreatic cancer cells*

Laetitia GIBART



Jury :

Rapporteurs

Morgan MAGNIN, PR, LS2N, Nantes
Sabine PERES, PR, LBBE, Lyon

Examineurs

Gilles BERNOT, PR, Université Côte d'azur, I3S, Sophia-Antipolis
Hélène COLLAVIZZA, MCF, Université Côte d'azur, I3S, Sophia-Antipolis
Jean-Paul COMET, PR, Université Côte d'azur, I3S, Sophia-Antipolis
Fariza TAHI, PR, IBISC, EVRY
Jean-Yves TROSSET, PR, Sup'biotech, Villejuif
Sophie VASSEUR, DR, CRCM Marseille

Université Côte d'Azur

Laetitia GIBART

Cycle de vie des modèles discrets et application à la régulation du métabolisme des cellules cancéreuses du pancréas

vi+198 p.

Cycle de vie des modèles discrets et application à la régulation du métabolisme des cellules cancéreuses du pancréas

Résumé

La modélisation des systèmes biologiques complexes ne bénéficie pas des méthodes et outils existants pour le développement des produits technologiques, comme en génie logiciel. La raison principale en est l'impossibilité de mettre en place une conception modulaire : en biologie, l'introduction d'un nouvel élément dans un modèle préalablement validé peut drastiquement remettre en cause son comportement. Cette thèse fournit néanmoins une méthode incrémentale de conception de modèles discrets, pour la biologie, qui possède certaines similarités avec les méthodes de développement de logiciel en spirale. L'objectif principal de l'approche est de préserver la faisabilité des preuves permettant de vérifier la non-régression à chaque enrichissement. Bien que la méthode proposée ait été développée dans le cadre du formalisme de René Thomas, elle peut s'adapter à tout autre cadre de modélisation discret. Elle comprend trois phases distinctes : l'analyse des besoins, l'enrichissement du modèle et sa validation par des méthodes formelles et enfin une phase de prédiction. L'étude de cas, conséquente, qui ancre ce travail, est un modèle de régulation du métabolisme à différents stades d'évolution de l'adénocarcinome ductal du pancréas (PDAC). Nous adoptons dans cette thèse une approche de modélisation abstraite en partant d'un modèle général, déjà conséquent, de la régulation du métabolisme eucaryote, que nous enrichissons pas à pas pour atteindre notre objectif : un modèle couvrant l'évolution de la cellule saine à la cellule agressive en passant par la cellule cancéreuse non agressive. On ne dispose a priori que de peu de connaissances moléculaires sur les dérégulations du métabolisme mises en jeu et le niveau d'abstraction de ce modèle est choisi pour orienter au mieux les biologistes dans leurs recherches et dans le choix de leurs hypothèses de travail.

Mots-clés : Conception de modèles complexes, Modélisation discrète, Réseaux de régulation biologique, Réseaux de R. Thomas, Cancer du pancréas, bio-informatique.

Discrete model life cycle and application to the regulation of pancreatic cancer cells

Abstract

The modelling of complex biological systems does not benefit from the methods and tools that exist for the development of technological products in software engineering. The main reason for this is the impossibility of implementing a modular design: in biology, the introduction of a new element in a previously validated model can drastically change its behaviour. Nevertheless, this thesis provides an incremental methodology, in order to design discrete models for biology, which has some similarities to software spiral development model. The principal goal is to preserve the feasibility of proofs allowing to verify the non-regression at each enrichment. Although the proposed method was developed within the framework of René Thomas' formalism, it can be adapted to any other discrete modelling framework. It includes three distinct phases: requirements analysis, model enrichment combined with validation by formal methods and finally a prediction phase. The substantial case study that anchors this work is a model of metabolic regulation at different stages of progression in pancreatic ductal adenocarcinoma (PDAC). In this thesis, we adopt an abstract modelling incremental approach, whose starting point is an already consequent general model of eukaryotic metabolic regulation. Our method allows us to enrich this initial model step by step to reach our aim, that is, a model covering the evolution from the healthy cell to the aggressive cell through the non-aggressive cancer cell. One has only little molecular *a priori* knowledge on the metabolic deregulation involved, and the level of abstraction of this model is chosen to best guide biologists in their research and in the choice of their working hypotheses.

Keywords: Complex model design, Discrete modelling, Biological regulatory networks, Thomas' regulatory networks, Pancreatic cancer, bioinformatics.

Table des matières

1	Introduction	1
	Notations	5
Un formalisme discret et formel pour modéliser les régulations		
2	Principaux formalismes pour modéliser les réseaux de régulations	7
2.1	Différentes représentations du métabolisme	8
2.2	Formalismes quantitatifs	9
2.3	Formalismes qualitatifs	15
3	Formalisme de René Thomas étendu	21
3.1	Graphe de régulation et paramètres	22
3.2	Dynamique d'un réseau de régulation	25
3.3	Analyse des traces par une logique temporelle	33
Cycles de vie pour les modèles formels de systèmes dynamiques		
4	Cycles de vie : du génie logiciel aux modèles de systèmes dynamiques	40
4.1	Méthodologie de création de modèles <i>de novo</i>	41
4.2	Méthodologie de modélisation par enrichissements successifs	44
5	Analyse du contexte, des principaux acteurs et des comportements attendus	48
5.1	Formalisation des régulations	49
5.2	Formalisation des phénotypes	50
5.3	Un langage pour les phénotypes courants	52
5.4	Formalisation de la matrice de phénotypes	55
6	Méthodes d'identification des paramètres discrets	58
6.1	Identification manuelle des paramètres	59
6.2	Identification formelle des paramètres	61
7	Méthodes de vérification de modèles	66
7.1	Méthode de validation globale de la matrice	68
7.2	Formalisation de la méthode par environnements	73
7.3	Automatisation de la méthode environnement par environnement	80
7.4	Conclusion	83

8	Modélisation par enrichissements successifs	85
8.1	Insérer un multiplexe	86
8.2	Insérer une variable	91
8.3	Ajouter un niveau à une variable préexistante	94
8.4	Scinder une variable ou un multiplexe préexistant	98
 Conception incrémentale d'un modèle de régulation du métabolisme dans l'adénocarcinome ductal pancréatique 		
9	Un modèle initial générique : régulation du métabolisme de cellules saines	107
9.1	La régulation du métabolisme et la glycolyse en aérobie	108
9.2	Composition du graphe de régulation	114
9.3	Formalisation des phénotypes : la matrice de validation	121
9.4	Spécifications détaillées : identification manuelle des paramètres	127
9.5	Validation de modèles à l'aide de <i>TotemBioNet</i>	138
10	Enrichissements pour un modèle de cellules cancéreuses pancréatiques non agressives	140
10.1	Contexte génétique et phénotypique du cancer	141
10.2	Les cofacteurs $NAD(P)H/NAD(P)^+$ dans le cancer	148
10.3	Les ROS comme marqueur de l'état cancéreux	153
11	Enrichissements supplémentaires pour les cellules agressives	159
11.1	Cancer agressif vs. non-agressif	160
11.2	Cycle d'enrichissement pour la méthionine	162
11.3	Cycle d'enrichissement pour les régulations mettant en cause l'agressivité	166
12	Conclusion et Perspectives	170
12.1	Les acquis de cette thèse	170
12.2	Aides à la prédiction	173
	Bibliographie	177
	Liste des figures	185
	Liste des tableaux	189
	Liste des définitions	192
	Liste des exemples	193
 Annexes 		
A	Application sur le modèle de la régulation du métabolisme	196
A	Notations pour le modèle de la régulation du métabolisme	196

B	Liste des paramètres du modèle de la régulation du métabolisme pour une cellule cancéreuse agressive.	198
---	--	-----

CHAPITRE 1

Introduction

Dans ce document nous développons un gros modèle formel de réseau de régulation biologique en utilisant la théorie de René Thomas. Cela nous a conduit à développer des méthodes de conception spécifiques, inspirées des méthodes de développement du génie logiciel, mais centrées sur le problème majeur des systèmes complexes : ils sont intrinsèquement non incrémentaux. L'essentiel de nos méthodes est donc guidé par la volonté de suivre l'évolution des propriétés globales du système, d'enrichissement en enrichissement, et de faciliter les preuves automatiques de non régression pour les propriétés à préserver.

La motivation pour concevoir ce gros modèle est l'étude du métabolisme des cellules cancéreuses du pancréas, depuis la cellule saine jusqu'à la cellule agressive, responsable de métastases létales, en passant par la cellule cancéreuse non agressive. Les enrichissements nécessaires du modèle de cellule saine ne se réduisent pas à deux étapes (cellule cancéreuse et cellule cancéreuse agressive) car les nouveaux concepts biologiques que chacune d'elles implique suffit à rendre très périlleuses les preuves de non régression. Ces deux types cellulaires donnent donc chacun lieu à bon nombre d'enrichissements élémentaires, selon notre méthodologie.

Le pancréas est un organe de forme allongée et triangulaire que l'on retrouve dans l'abdomen derrière l'estomac. Cet organe mesure environ 15 cm de long et comporte trois parties : la tête, le corps et la queue. Le pancréas possède deux fonctions importantes : la sécrétion de sucs digestifs et la sécrétion d'hormones pour la régulation de la glycémie. Chaque fonction y est exercée par un type cellulaire différent [2]. Les cellules exocrines produisent plusieurs enzymes et des sucs digestifs qui jouent un rôle clé dans la digestion des aliments. Ces sucs sont ensuite acheminés dans la lumière de l'intestin par des canaux. Les cellules endocrines quant à elles produisent des hormones qui agissent « à distance » et sont déversées directement dans le sang [Pandol, 2011].

Lorsqu'un cancer se développe dans le pancréas, il se localise dans près de 90% des cas dans les cellules exocrines qui bordent les canaux conduisant le suc pancréatique vers l'intestin. C'est l'**adénocarcinome canalaire du pancréas**. (PDAC) [1]. D'autres formes plus rares existent comme les carcinomes mucineux pour n'en citer qu'un seul. Cependant, en plus d'être moins fréquents, ces autres types de cancer du pancréas ont un pronostic vital beaucoup moins engagé que l'adénocarcinome canalaire pancréatique [Fesinmeyer et al., 2005].

Maladie plus fréquente dans les pays industrialisés avec de 466 003 décès par an dans le monde, le cancer du pancréas est classé au 12e rang des cancers les plus fréquents d'après le site « global cancer observatory », dont les données proviennent de l'organisation mondiale de la santé, [6]. L'âge moyen du diagnostic de la maladie est de 75 ans. Malgré sa relativement faible incidence dans l'ensemble des maladies, 80 à 90% des patients atteints de ce type de cancer en décèdent. La cause de ce constat est le diagnostic très tardif de la maladie puisque les phases initiales du développement du cancer sont souvent asymptomatiques [7]. Bien souvent, lors du dépistage, le

cancer a déjà envahi d'autres organes ce qui fait que les traitements classiques utilisés contre le cancer, comme la chirurgie, ne sont plus envisageables. La chimiothérapie et les radiothérapies sont aussi d'une efficacité limitée lorsque le cancer est à un stade de développement aussi avancé.

Pour pallier la forte mortalité du cancer, une des stratégies de recherche est de développer des techniques de dépistage capables de diagnostiquer très précocement ce type de cancer. Cependant, un dépistage précoce, resterait un moyen préventif qui ne sauverait pas tous les patients atteints de la maladie. Certains chercheurs s'appuient donc sur un second angle d'attaque : la compréhension des différents mécanismes de l'évolution du cancer, et ce, dans le but de trouver des solutions pour bloquer les mécanismes évolutifs de la maladie, évitant ainsi que le cancer envahisse d'autres organes. C'est sur cette idée que Sophie Vasseur (au centre de recherche en cancérologie de Marseille [5]) tente d'élucider les mécanismes au niveau moléculaire de l'évolution de l'adénocarcinome canalaire du pancréas. Cette équipe est spécialisée dans l'analyse du *métabolisme* de la cellule pancréatique atteinte du cancer [4].

Le métabolisme cellulaire est l'ensemble des réactions biochimiques qui permettent de maintenir l'équilibre de la cellule pour sa survie. Les réactions sont regroupées en voies métaboliques. Ces voies vont être plus ou moins actives et finement régulées en fonction de l'état énergétique de la cellule. Lorsque la cellule est en manque d'énergie (pour se diviser ou se mouvoir), une régulation positive des différentes voies métaboliques permettant la création d'énergie se met en place. A l'inverse lorsque la cellule a suffisamment d'énergie, elle va la stocker sous forme de macromolécules et réguler négativement les voies qui vont libérer de l'énergie [Berg et al., 2002, DeBerardinis and Thompson, 2012, Nelson and Cox, 2012, Koolman and Röhm, 2012]. De manière générale dans le cancer, le métabolisme de la cellule est impacté, certaines voies sont dérégulées en comparaison d'une cellule saine. La découverte des mécanismes de régulation nécessaires pour l'évolution du cancer vers des stades agressifs est donc un point clef pour la mise en place d'une stratégie thérapeutique [Hsu and Sabatini, 2008, Cairns et al., 2011].

Un des problèmes principaux de cette approche est de faire face à la multitude des mécanismes. En effet, le nombre de réactions dans le métabolisme ainsi que leurs interconnexions est tel qu'il est impossible d'explorer l'ensemble des réactions pouvant être importantes. La majeure partie des études se fait donc de voie métabolique en voie métabolique en éliminant celles qui sont expérimentalement démontrées sans intérêt.

C'est en ce point que la modélisation de systèmes biologiques peut avoir son intérêt [Kitano, 2002]. En effet, créer un modèle dynamique capable de capturer les régulations générales du métabolisme permettrait d'orienter les biologistes, *via* des « prédictions », vers les voies métaboliques qui pourraient avoir un intérêt pratique. C'est en partie le sujet du projet appelé PAIR-pancréas « criblage métabolique du cancer du pancréas métastatique », auquel j'ai contribué pendant ces trois ans de thèse.

La modélisation de systèmes dynamiques a pour but de comprendre et formaliser le comportement des systèmes complexes dans le temps. Le métabolisme cellulaire est un système dynamique puisque l'ensemble des éléments qui le constituent évoluent au cours du temps. Différents cadres de modélisation existent pour représenter les systèmes dynamiques. Chacun des cadres de modélisation offre des points de vue différents vis-à-vis de la question biologique ini-

tiale [Gombert and Nielsen, 2000, Vaitheesvaran et al., 2015, Reiser et al., 2001]. Nous nous intéresserons au formalisme de René Thomas [Thomas, 1973a] qui propose un cadre de raisonnement discret, plus aisé à assister par ordinateur.

Communément, la construction d'un modèle se base uniquement sur l'intégration monolithique de connaissances biologiques qui permettent de construire un réseau dynamique. Cette technique peut s'apparenter dans notre problématique à un cycle de développement en « V » [Printz, 2006], très connu dans le domaine du génie logiciel. Plus le réseau est grand et les interconnexions entre les différents éléments importantes, plus le temps de conceptualisation du modèle est long à tel point qu'on ne remplit plus le but de venir en aide aux biologistes. De plus, lors de la conception d'un modèle, les outils formels utilisés pour la conception atteignent leurs limites. Nous nous sommes donc intéressés à améliorer la démarche de conception pour ces différents aspects. Nous avons donc dû proposer de nouvelles méthodes de travail pour la conception de tels modèles.

Dans un premier temps, la solution évidente est de réutiliser un modèle déjà conçu et de l'adapter à la nouvelle problématique et nous utiliserons un modèle représentant la régulation du métabolisme de la cellule.

L'adaptation d'un réseau passe par l'analyse et la prise en compte de nouvelles informations ou d'expériences. Cela peut nécessiter l'ajout de différents composants. En toute généralité, ces ajouts peuvent entraîner un changement drastique des comportements établis. C'est pourquoi un travail sur les méthodes de développement de modèle, et de formalisation d'information, est nécessaire avant de se focaliser sur la conception d'un modèle de la régulation du métabolisme dans le contexte du cancer du pancréas. En pratique le travail sur la méthodologie de développement de modèles implique aussi des ajustements techniques pour l'utilisation des outils formels pour la modélisation.

Finalement, pour répondre au mieux aux différents questionnements des biologistes, nous prenons, dans ce manuscrit, le parti de proposer une nouvelle méthode de développement de modèles qui vient en complément pour aider la conception de modèles de grande taille tout en évitant toute régression des spécifications initiales du modèle.

Le développement de modèle se base globalement sur les mêmes principes que le développement d'outils logiciels, puisque l'on souhaite produire un modèle qui repose sur les besoins des biologistes, et sur des spécifications correspondant aux connaissances sur le système étudié. Pour cette raison, nous faisons le choix de nous inspirer de la méthode de développement en spirale, déjà existante en génie logicielle, pour le développement de notre nouvelle méthodologie de construction de modèles. La mise en place de cette nouvelle méthodologie nécessite quelques adaptations que nous décrivons et que nous utilisons par la suite pour la conception d'un modèle de la régulation du métabolisme dans le cancer du pancréas.

Dans ce contexte, nous avons fait le choix de séparer le document en trois parties : l'état de l'art, les travaux théoriques sur le cadre de modélisation et les travaux applicatifs sur la modélisation de la régulation du métabolisme dans le cancer du pancréas.

La première partie permet à la fois de positionner et de comprendre le cadre de modélisation de René Thomas, cadre sur lequel repose l'ensemble des travaux de thèse. Elle se découpe en deux chapitres : le **Chapitre 2** décrit brièvement les principaux formalismes (quantitatif et qualitatifs) pour modéliser un réseau métabolique. Le **Chapitre 3** présente la démarche adoptée pour la modélisation d'un nouveau réseau de régulation, en utilisant le cadre de modélisation de R. Thomas.

La seconde partie vise à décrire proprement les différentes méthodes pour développer et enrichir un modèle. Nous introduisons dans le **chapitre 4** les deux méthodes que l'on va utiliser successivement pour développer un modèle. La première consiste, à partir des simples connaissances biologiques que l'on formalise, à créer un contexte de modélisation formel. La seconde méthode se base sur un contexte de modélisation préétabli et y ajoute de nouvelles connaissances. La description de ces deux grandes méthodologies complémentaires permet de poser le cadre et comprendre les autres chapitres de cette seconde partie. Dans le **chapitre 5** nous faisons le lien entre les connaissances biologiques et leurs formalisations pour la description d'un nouveau contexte de modélisation. Le **chapitre 6** discute des différentes méthodes pour identifier les paramètres. Deux méthodes sont présentées : la première est basée sur les connaissances de la dynamique locale avec l'aide des biologistes et la seconde repose sur l'utilisation de *TotemBioNet* un outil permettant l'utilisation intense de méthodes formelles. Dans le **chapitre 7** : une fois construit, le contexte de modélisation doit être validé, que l'on soit dans un contexte d'enrichissement ou pour la description d'un nouveau modèle *ex nihilo*. Nous proposons deux méthodes de vérification formelle des paramétrisations *via TotemBioNet*. Une première est en « force brut » et permet la vérification de l'ensemble du contexte en un lancement de *TotemBioNet*; une seconde se base sur une approche environnementale et automatisée par l'ajout d'un module à *TotemBioNet*. Le **chapitre 8** se base sur la méthodologie d'enrichissements et propose différentes solutions formelles pour enrichir un contexte de modélisation en fonction des différentes connaissances biologiques à ajouter. Une fois formalisés, ces nouveaux contextes de modélisation sont validés.

Enfin, la troisième et dernière partie applique les différentes méthodes décrites dans la partie précédente. C'est une partie applicatrice des différents modèles développés pour représenter la régulation du métabolisme dans l'adénocarcinome ductal pancréatique.

Le **chapitre 9** décrit le modèle de base et créé à l'aide de la première méthode de modélisation en « V », en se basant sur les connaissances biologiques générales. Le modèle décrit la régulation du métabolisme dans le cadre de la glycolyse en aérobie. L'idée initiale de ce modèle est de proposer une référence générale qui pourra ensuite être adaptée à différentes problématiques. Le **chapitre 10** propose, à travers différents enrichissements incrémentaux, la conception d'un modèle de régulation du métabolisme dans un cadre cancéreux en se basant sur les connaissances de l'équipe de Sophie Vasseur. Le **chapitre 11** propose de la même manière différents enrichissements incrémentaux du modèle présenté au chapitre 10, et met en place un prototype de modèle de régulation du métabolisme dans un cadre cancéreux agressif. Le **chapitre 12** constitue une conclusion générale à tous ces travaux. Il permet tout d'abord de prendre un peu de recul sur les aspects techniques présentés aux chapitres précédents, d'en discuter tous les apports théoriques et méthodologiques. L'expérience acquise au cours de ces développements ouvre de nouvelles perspectives à la fois du point de vue de la conception de modèles assistée par les méthodes formelles et du point de vue de l'exploitation des ensembles de modèles définis à chaque étape de notre développement en spirale.

Bien que nous proposons le développement d'une méthodologie d'enrichissement dans le cadre de René Thomas, la démarche qui vise enrichir des modèles déjà valides plutôt de manière à faciliter les preuves de non régression a une portée beaucoup plus large. En effet, ce concept peut théoriquement s'adapter à l'ensemble des cadres de modélisation et ouvre donc de nombreuses perspectives.

Notations

Modèle de régulation du métabolisme

<i>ADP</i>	Adenosine bi-phosphate
<i>AMP</i>	Adenosine mono-phosphate
<i>ARN</i>	Adénosine ribonucléique
<i>ATP</i>	Adenosine tri-phosphate
$\beta - ox$	β -oxydation , voie de dégradation des acides gras
<i>CDK2A</i>	Cyclin dependent kinase inhibitor 2A
<i>EMT</i>	transition épithélio mésenchymateuse
<i>FA</i>	Acides gras (Fatty acids)
<i>FAD</i>	Flavine adénine dinucléotide
<i>KRAS</i>	V-Ki-ras2 Kirsten rat sarcoma viral oncogene homolog
<i>GDP</i>	Guanosine diphosphate)
<i>GSH</i>	Glutathion modlécule permettant la réduction des ROS
<i>GSSH</i>	Glutathion sous forme oxydé
<i>GOT1</i>	Enzyme appelé Aspartate aminotransferase surexprimé dans le PDAC
<i>GPx</i>	Enzyme de la détoxification des ROS appelée peroxydase
<i>GR</i>	Enzyme appelé glutathion réductase
<i>GTP</i>	Guanosine triphosphate)
<i>ME1</i>	enzyme appelé NADP-dependent malic surexprimée dans le PDAC
<i>NAD</i>	Nicotinamide adénine dinucléotide
<i>NOX</i>	NADPH oxydase enzyme créatrice de ROS
<i>O₂</i>	Oxygène
<i>PDAC</i>	Adénocarcinome ductal du pancréas
<i>PPP</i>	Voie métabolique des pentoses phosphates
<i>PRx</i>	Péroxydase, enzyme de détoxification des ROS
<i>ROS</i>	Espèces réactives à l'oxygène
<i>TP53</i>	Tumor protein 53, protéine du cycle cellulaire
<i>SMAD4</i>	Mother against decapentaplegic homolog 4, gène impliqué dans le cycle cellulaire
<i>SOD</i>	Superoxyde dismutase enzyme de détoxification des ROS

**Un formalisme discret et formel pour
modéliser les régulations**

Principaux formalismes pour modéliser les réseaux de régulations

Ce chapitre présente différents formalismes pour modéliser un réseau métabolique. Puisque notre travail se focalise sur la modélisation des comportements nous allons nous limiter à la description des différents formalismes capables de représenter la dynamique du métabolisme. Notamment, ce chapitre propose une description de formalismes quantitatifs tel que les EDOs (équations différentielles ordinaires) et l'analyse de flux. Nous y décrivons aussi certains formalismes qualitatifs, tels que, les réseaux de Petri, les modèles logiques basés sur des règles et les réseaux booléens et multivalués.

2.1	Différentes représentations du métabolisme	8
2.2	Formalismes quantitatifs	9
2.2.1	Formalismes dédiés aux flux métaboliques	9
2.2.2	EDO : modélisation des réactions élémentaires	13
2.3	Formalismes qualitatifs	15
2.3.1	Réseaux de Petri	15
2.3.2	Formalismes à base de règles de transformation	17
2.3.3	Réseaux booléens	18

De nombreux formalismes sont disponibles pour modéliser les systèmes biologiques. Le choix du formalisme approprié est déterminé pour répondre au mieux aux questionnements des biologistes. Ce choix dépend notamment du niveau de détail de la problématique, et du type de réponse que l'on cherche à obtenir. Dans notre cas nous cherchons des réponses à propos des **régulations** du métabolisme impliquées dans le cancer. Il est donc primordial d'avoir un formalisme capable de capturer ces informations. Commençons par décrire dans la section suivante deux méthodes principales de description du métabolisme.

2.1 Différentes représentations du métabolisme

Nous pouvons dégager deux grandes classes d'approches pour la représentation de l'évolution du métabolisme.

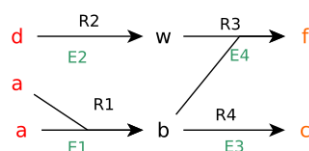


Figure 2.1 – Exemple d'un réseau métabolique. En rouge sont les métabolites d'entrée, en orange ceux de sortie. En vert sont les enzymes des réactions. R1, R2, R3, R4 nomment les réactions du réseau métabolique.

D'une part le métabolisme peut être représenté d'un point de vue fonctionnel : on y décrit chaque réaction moléculaire entre les différents métabolites comme exposé figure 2.1. Dans ce cas, on décrit les éléments nécessaires pour que les réactions se produisent, enzyme et cofacteurs compris.

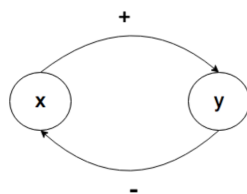


Figure 2.2 – Exemple d'un graphe de régulation. Les entités X et Y représentent des éléments abstraits de voies métaboliques.

D'autre part on peut décrire l'évolution du métabolisme en mettant l'accent sur les liens de régulation entre les entités (voir figure 2.2). Dans ce cas on décrit les actions de régulation qu'ont les entités entre elles sans indiquer les règles de transformations réactionnelles. Prenons un exemple. Si X est transformé par une réaction en Y , cela signifie que X est activateur de Y et comme X disparaît quand Y apparaît alors Y régule négativement X . On peut ensuite représenter cet exemple de description par un graphe de régulation, ce qui est donné figure 2.2.

Ces deux manières de représenter des réseaux métaboliques seront utilisées dans la suite de ce chapitre à titre d'exemple pour comprendre les différents formalismes exposés.

Les formalismes les plus utilisés pour modéliser le métabolisme se basent sur la description des réactions (figure 2.1) de manière quantitative, soit par des méthodes d'analyse de flux (EFM et FBA) soit par les équations différentielles ordinaires (EDO). Nous décrirons dans un second temps certaines méthodes de modélisation qualitative, qui permettront plus de souplesse et facilitent la représentation des voies métaboliques du point de vue des régulations. La présentation de chaque formalisme sera séparée en deux parties principales :

- La description des bases sur lesquelles repose le formalisme,
- et la description des avantages et inconvénients du formalisme décrit vis à vis de la modélisation de la régulation du métabolisme dans le cadre de l'évolution du cancer du pancréas.

2.2 Formalismes quantitatifs

La modélisation quantitative a pris beaucoup d'ampleur depuis les années 2000 avec l'explosion des données omiques. L'analyse expérimentale comprenant les mesures des concentrations d'enzymes, des activités d'enzymes, de constantes de vitesse de réaction et de flux métaboliques sont devenues des standards. De nombreux formalismes ont été proposés dans la littérature, mais nous accordons une attention particulière aux deux méthodes suivantes : les EDOs et les Analyses de flux (tel que FBA ou EFM) [Wang, 2013, Orth et al., 2010, Rui et al., 2015].

Dans le cadre de la modélisation du métabolisme les formalismes les plus utilisés de nos jours restent incontestablement les formalismes basés sur l'analyse des flux : FBA (Flux Balance Analysis) et EFM (Elementary Flux Mode) et leurs variantes. Nous commencerons donc par la description de ces formalismes.

2.2.1 Formalismes dédiés aux flux métaboliques

2.2.1.1 Généralités

Ces méthodes se placent au niveau moléculaire, c'est à dire que les éléments de base d'un modèle représentent des molécules. Dans un tel contexte, un réseau métabolique fait donc référence à une série de réactions biochimiques inter-connectées comme représentées en figure 2.1. Dans ces formalismes, la stœchiométrie et les vitesses sont des notions très importantes pour l'analyse de ces réactions d'un point de vue quantitatif. La stœchiométrie fait référence à la relation quantitative entre les réactifs et les produits : les coefficients stœchiométriques indiquent la quantité de chaque substrat par rapport à la quantité de produits obtenus. Notons que dans toute réaction chimique, la loi de la conservation de la masse s'applique. Cela signifie qu'il y a une relation d'équilibre entre la quantité de produit formé et de réactif consommé. L'équilibre précité est défini par l'intermédiaire des coefficients stœchiométriques d'une réaction chimique. Par exemple, pour la réaction $r1$, qui transforme a en b donné figure 2.1, il faut $2a$ pour donner $1b$. Les coefficients stœchiométriques correspondent respectivement à 2 et 1 pour la réaction précitée.

Une fois que nous disposons de l'ensemble des réactions et des coefficients stœchiométriques (première étape), l'étape suivante consiste à déterminer la vitesse de flux de chaque réaction, c'est-à-dire la vitesse à laquelle elles se produisent.

Nous nous concentrons sur l'exemple figure 2.1 pour illustrer l'utilisation de la stœchiométrie et de la vitesse de flux requis dans FBA, EFM. Quatre réactions biochimiques y sont représentées : $r1$ que l'on note $r1 : 2a \rightarrow b$ dont on a déjà parlé dans le paragraphe précédent, $r2$, $r3$ et $r4$. La réaction $r2$ indique la transformation de d en w ($r2 : d \rightarrow w$), la réaction $r3$ la transformation de w et b en f ($r3 : w + b \rightarrow 2f$) et la réaction $r4$ de b en c , noté $r4 : b \rightarrow c$.

Si on regarde la structure de la figure 2.1, on constate qu'un métabolite peut à la fois être un substrat et un produit, c'est le cas de w qui participe à la réaction $r2$ en temps que produit et à $r3$ en substrat.

L'objectif des modèles en analyse de flux est de contrôler rigoureusement les flux, autrement dit les vitesses des réactions pour optimiser la production de certains métabolites particuliers. C'est pour cette raison que ce type de modélisation est très utilisé dans l'industrie, en production [Boghigian et al., 2010]. Dans notre exemple, nous supposons que nous cherchons à maximiser la production de c .

Modèle en analyse de flux En analyse de flux chaque système métabolique peut partiellement être représenté par une matrice dite de "stœchiométrie", de dimension $M \times R$ avec M le nombre de métabolites et R le nombre de réactions. Dans la représentation matricielle, les métabolites forment les lignes et les réactions forment les colonnes. Une valeur 0 dans la matrice signifie que la réaction r n'implique pas le métabolite associé, que ce soit sous forme de produit ou de substrat. une valeur supérieure à 0 indique que le métabolite est un produit de la réaction. A l'inverse si la valeur est négative cela indique que le métabolite est un substrat qui, par définition sera consommé pendant la réaction.

La matrice S équation (2.1), représente la matrice de "stœchiométrie" pour notre exemple figure 2.1, où $S_{x,ri}$ correspond à l'intersection d'une ligne x et d'une colonne ri dans la matrice. Pour la réaction $r1$, $2a$ sont consommés, on note donc -2 dans $S_{a,r1}$ et b est produit, on note donc 1 dans $S_{b,r1}$. Pour la réaction $r2 : d \rightarrow w$, d est consommé donc $S_{d,r2} = -1$ et w produit donc $S_{w,r2} = 1$. Pour $r3 : w + b \rightarrow 2f$, $S_{w,r3} = -1$, $S_{b,r3} = -1$ et $S_{f,r3} = 2$. Enfin pour, $r4 : b \rightarrow c$, $S_{b,r4} = -1$ et $S_{c,r4} = 1$.

$$\mathbf{S} = \begin{matrix} & \begin{matrix} r1 & r2 & r3 & r4 \end{matrix} \\ \begin{bmatrix} -2 & 0 & 0 & 0 \\ 1 & 0 & -1 & -1 \\ 0 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 2 & 0 \end{bmatrix} & \begin{matrix} a \\ b \\ d \\ w \\ c \\ f \end{matrix} \end{matrix} \quad (2.1)$$

Nous pouvons passer à l'introduction d'un vecteur v correspondant au différentes vitesses de réaction. L'ensemble des vitesses de réaction sont *a priori* non connues, $r1$, $r2$, $r3$ et $r4$, auront pour vitesses respectives v_1, v_2, v_3 et v_4 . Le vecteur \vec{v} s'écrit :

$$\vec{v} = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{pmatrix} \quad (2.2)$$

En faisant ensuite le produit scalaire de $S \times \vec{v}$, on obtient le vecteur des vitesses, appelé aussi flux, résultant des métabolites. De cette équation on déduit par exemple que le flux de a est $-2v_1$.

$$\mathbf{S} * \vec{v} = \begin{bmatrix} -2 & 0 & 0 & 0 \\ 1 & 0 & -1 & -1 \\ 0 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 2 & 0 \end{bmatrix} \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{pmatrix} = \begin{bmatrix} -2v_1 \\ v_1 - v_3 - v_4 \\ -v_2 \\ v_2 - v_3 \\ v_4 \\ 2v_3 \end{bmatrix} \quad (2.3)$$

Dans le système nous avons deux types de métabolites, les métabolites internes b et w qui sont produits et substrats du système représenté et les métabolites externes qui sont à l'entrée ou à la sortie du système : a , d , f , et c .

La majorité du temps les études en biologie se font alors que le système est dans un état stationnaire. Or si l'on considère cet état, la concentration de chaque métabolite à l'intérieur du système peut être considéré comme constant : le flux d'un métabolite interne est donc par conséquent nul ($S \cdot \vec{v} = 0$). Cette affirmation ne s'applique pas aux métabolites externes. En effet, par exemple les métabolites a et d sont des métabolites d'entrée de la voie. Leur quantité ne peut pas tendre vers 0 car il ne sont que consommés par le système, le flux de ces métabolites n'est donc par conséquent pas égal à 0. Les métabolites f et c , ne sont que produit par le système : leur quantité ne fait qu'augmenter au cours du temps, ce qui indique que leur flux n'est pas non plus égal à 0.

En supposant pour notre exemple que les flux des métabolites internes son nuls on obtient :

$$\begin{bmatrix} v_1 - v_3 - v_4 \\ v_2 - v_3 \end{bmatrix} \begin{matrix} b \\ w \end{matrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

En utilisant $S \cdot \vec{v} = 0$ sur les métabolites internes, on impose certaines contraintes sur les vitesses de réactions. Dans notre exemple, des métabolites internes on déduit les deux équations suivantes : $v_1 - v_3 - v_4 = 0$ et $v_2 - v_3 = 0$. Nous pouvons dire que que $v_1 = v_3 + v_4$ et $v_2 = v_3$. Cela permet de réduire le nombre de solutions possibles, cependant en ce point l'ensemble des solutions reste très grand.

Jusqu'ici nous avons décrit le principe d'analyse des flux de manière générale sur lequel repose à la fois les modèles FBAs et EFMs. Passons maintenant aux différences notables entre ces deux formalismes.

Particularités du FBA Le FBA [[Schuster and Hilgetag](#),] peut donner un grand nombre de solutions mathématiquement acceptables au problème de l'état d'équilibre $S \cdot \vec{v} = 0$. Cependant les solutions d'intérêt biologique sont celles qui produisent les métabolites recherchés dans la bonne proportion. La fonction objectif définit la proportion de ces métabolites. Le Flux balance analysis est une analyse de flux métabolique utilisée pour déterminer les vitesses permettant de maximiser ou minimiser la fonction objective [[Raman and Chandra, 2009](#)]. Par exemple, lors de la modélisation de la croissance d'un organisme, la fonction objectif est généralement définie comme la biomasse. Mathématiquement, on va pondérer notre réseau de réactions selon l'importance des réactions et optimiser le résultat produit en biomasse dans la cellule. Selon ce qu'on veut étudier, on

ajuste les différents paramètres des réactions, afin de maximiser la fonction objectif. La fonction objectif peut être déterminée pour maximiser ou minimiser la production d'autres éléments que les métabolites liées à la biomasse. Au final le formalisme FBA s'appuie sur des logiciels d'optimisation pour donner la/les solutions optimale(s) pour un réseau métabolique donné, en maximisant ou minimisant la production d'un métabolite ou ensemble de métabolites.

Particularités du EFM L'approche appelé elementary flux mode (EFM) [Varma and Palsson,] est une variante au FBA proposé par [Schuster et al., 2000]. Ce cadre de modélisation partage le même principe mathématique (premières étapes) avec FBA ; c'est-à-dire, premièrement, on construit le réseau avec un ensemble initial de réactions et ensuite on construit la matrice de stoechiométrie et on se considère à l'état stationnaire. L'EFM introduit une méthode systématique d'extraction des voies (suites de réactions) d'un réseau métabolique complexe qui sont « élémentaires » et ces voies sont appelées *modes élémentaires*. Ils sont choisis pour engendrer la totalité des régimes possibles (par combinaison linéaires). Cet ensemble de générateur n'est pas unique cependant.

2.2.1.2 Avantages et inconvénients

Ces méthodes reposent sur des connaissances simples telle que le réseau et la stœchiométrie. Ils sont relativement simples dans leur utilisation et permettent d'obtenir des résultats directement à partir de données brutes issues de bases de données telles que KEGG¹. Cependant comme l'analyse de flux se base sur la description au niveau moléculaire la représentation du métabolisme cellulaire dans son entièreté requière de nombreuses connaissances détaillées. Même en se reposant sur les bases de données, le temps de curation de l'ensemble des informations disponibles dans le cadre du cancer du pancréas reste relativement restreint, ce qui complexifie le processus de modélisation avec ces méthodes.

De plus, particulièrement pour FBA, bien qu'il soit très utilisé pour ces performances dans l'industrie en production pour maximiser la production de métabolites d'intérêts [Raman and Chandra, 2009], il ne répond que marginalement à l'étude des régulations entre les voies métaboliques, FBA peut donc pas être utilisé pour répondre à notre problématique. En outre, même si EFM peut mettre évidence les voies d'intérêt, il ne permet pas directement la représentation des régulations (causalité et mécanistes), et par extension n'est pas capable de répondre à la problématique initiale qui motive la modélisation du métabolisme dans le cancer du pancréas. Pour avoir une analyse des régulations il faudrait simuler plusieurs configurations possibles de réseaux et faire une analyse comparative des différences dans l'ensemble des configurations testées pour tenter une interprétation mécanistique. Nous ne reviendrons pas sur ce type de méthodes, car nous souhaitons nous focaliser sur les régulations du métabolisme.

1. KEGG est une base de donnée en ligne de voies métaboliques enregistrées sous différents formats, voir <https://www.genome.jp/kegg/>

2.2.2 EDO : modélisation des réactions élémentaires

2.2.2.1 Généralités

Les réactions biologiques peuvent être aussi décrites par des équations différentielles linéaires ou non linéaires, ordinaires, dont les solutions fournissent des informations sur la dynamique des processus étudiés [Resat et al., 2009]. Ce formalisme dynamique repose sur un système d'équations avec des coefficients à déterminer représentant les paramètres dynamiques des réactions en jeu. L'idée repose sur la description de l'évolution des concentrations des différents substrats, enzymes, produits du système par des équations différentielles. Le succès des équations différentielles ordinaires est dû à leurs capacités à décrire l'évolution temporelle des composants d'un système. Elles ont été utilisées dans diverses applications en ingénierie et en sciences naturelles.

2.2.2.2 EDO pour représenter l'évolution des métabolites

On reprend l'exemple des réactions métaboliques, que l'on a déjà utilisées pour définir les formalismes basés sur les flux. Après avoir obtenu la matrice de stoechiométrie et le vecteur de flux (notés respectivement S et \vec{v}), le système (x_i représente les métabolites) peut être exprimé par :

$$\left(\frac{d\vec{x}_i}{dt}\right) = S \cdot \vec{v}$$

Le taux de variation de chacun des métabolites est donné par une fonction linéaire des flux affectant le métabolite (un flux est négatif si le métabolite est consommé). Si l'on reprend les équations linéaires proposées pour l'analyse des flux, le système d'équations devient :

$$\begin{aligned} \frac{da}{dt} &= -2v_1 \\ \frac{db}{dt} &= v_1 - v_3 - v_4 \\ \frac{dd}{dt} &= -v_2 \\ \frac{dw}{dt} &= v_2 - v_3 \\ \frac{dc}{dt} &= v_4 \\ \frac{df}{dt} &= 2v_3 \end{aligned}$$

A l'état d'équilibre, comme pour les méthodes d'analyses de flux, les métabolites internes sont constants. Par conséquent, les métabolites b et w , qui sont métabolites internes, peuvent être exprimés comme EDO :

$$\begin{aligned} \frac{db}{dt} &= v_1 - v_3 - v_4 \\ \frac{dw}{dt} &= v_2 - v_3 \end{aligned}$$

Après avoir obtenu une représentation mathématique du modèle en termes d'EDO, de nombreuses méthodes numériques sont disponibles pour explorer le système. Pour chaque métabolite, on peut par exemple tracer graphiquement son évolution sur temps.

2.2.2.3 EDO pour représenter les régulations d'un système biologique

Dans l'exemple ci-dessus nous avons en partie repris la formalisation des réactions métaboliques en terme de flux mais les EDO permettent aussi de représenter un système de régulations. Écrivons le système d'équations pour représenter les régulations de l'exemple de la figure 2.2.

Voyons maintenant l'utilisation des équations différentielles pour représenter les *régulations* géniques dans lesquelles l'expression des gènes change avec le temps. En figure 2.2 nous avons deux régulations : X active Y et Y inhibe X . Les fonctions représentant l'action de X sur Y , f_y^x , et l'action de Y sur X , f_x^y sont sigmoïdales.

la vitesse de production du gène X dépend de la fonction de l'action de Y sur X , notée $f_x^y(y)$, de la vitesse minimale d'expression du gène sans activation, noté k_0^x et de la vitesse de dégradation représenté par φ^x . il en est de même pour Y .

Au final, l'évolution de la concentration des gènes X et Y peut être exprimé par :

$$\frac{dx}{dt} = k_0^x + f_y^x(y) - \varphi^x x \quad \frac{dy}{dt} = k_0^y + f_x^y(x) - \varphi^y y \quad (2.4)$$

où :

- f_x^y (resp. f_y^x) : représente l'action de x (resp. y) sur y (resp. x).
- φ^x représente la dégradation du gène x
- k_0^x représente la vitesse de production minimale lorsque x est inhibé.

2.2.2.4 Avantages et inconvénients

L'avantage principal des EDO, est leur capacité à englober des phénomènes biologiques plutôt hétérogènes (réactions métaboliques, régulations, voies de signalisation, etc.). Le caractère adaptatif du formalisme permet de représenter différents systèmes ne possédant pas les mêmes spécificités. En effet, ce formalisme permet la représentation de phénomènes à différentes échelles (différents niveaux d'abstractions) comme par exemple au niveau environnemental avec des modèles proies prédateurs, au niveau individuel/cellulaire pour représenter de la régulation entre les éléments ou au niveau moléculaire dans un cadre unique et homogène. Le couplage des régulations ainsi que des réactions enzymatiques peut donc être aisément codé en équations différentielles (ce qui est primordial dans la représentation de la régulation du métabolisme), simplement en « accumulant » les équations dans un système d'équations.

Comme ce formalisme peut s'adapter pour représenter des phénomène à différentes échelles, il est possible d'utiliser l'abstraction pour représenter le métabolisme à gros grain en proposant un système d'équations qui regroupe les réactions et les interactions dans une même entité qui évolue au cours du temps. Au lieu de représenter les réactions biochimiques directement on serait capable de représenter les éléments par exemple au niveau des voies métaboliques. Cependant on entre ici dans des considérations complexes d'approximations correctes.

Malheureusement, l'adaptabilité de ce formalisme est aussi un handicap. En effet, dans notre cas on cherche à représenter les réactions au sein du métabolisme cellulaire régulé par l'évolution du cancer. Il faut donc aussi ajouter les régulations dues à l'évolution du cancer, ce qui augmente

considérablement le nombre de paramètres dynamiques à prendre en compte. Or, malgré de nombreuses méthodes d'analyse de modèles, l'ensemble des paramètres dynamiques reste quasiment impossible à évaluer sur de très gros modèles, souvent à cause du manque d'informations biologiques, mais aussi car le domaine (\mathbb{R}) dans lequel on fait cette recherche est trop grand.

De plus, si le modélisateur arrive à codifier l'estimation de l'ensemble des paramètres dynamiques, les données mécanistiques pour représenter toutes les réactions au sein du métabolisme ne sont pas toutes connues en détails (par exemple, la vitesse de réaction, quantité de produit etc.). Ce qui rend complexe l'élaboration adaptative des formules précises pour chaque réaction ou ensemble de réactions, même s'il existe des généralisations pour pallier tous ces problèmes de calibration.

En bref, les modèles quantitatifs ont l'avantage de pouvoir décrire l'évolution d'un système en se plaçant au niveau le plus bas, proche des explications mécanistiques (réactions métaboliques). Par contre dans une approche très intégrée, cette capacité de description de tous les événements semble plutôt être un frein, puisque, d'une part les formalismes demandent une connaissance quasi exhaustive des phénomènes, et d'autre part, l'identification des paramètres est l'étape limitante.

2.3 Formalismes qualitatifs

Les méthodes de modélisation qualitatives reposent moins directement sur la précision des mesures biologiques expérimentales. En effet, les quantités exactes des différents métabolites ne nous importent pas. Ce qui est important, ce sont les chaînes de causalité (a active b et c inactive d , par exemple). On a donc la possibilité de raisonner sur des connaissances ou hypothèses qualitatives pour se placer à un niveau plus *abstrait*, c'est l'un des points forts des cadres de modélisation qualitatifs. Avec ces formalismes on s'attarde d'avantage à l'exploration et l'établissement de propriétés générales du système étudié.

Dans cette section nous discuterons de différents formalismes qualitatifs pouvant s'adapter au métabolisme tel que les réseaux de Petri, les formalismes basés sur les règles de transformation des réseaux booléens, et multivalués.

2.3.1 Réseaux de Petri

Les réseaux de Petri sont dédiés à l'étude de la concurrence et sont apparus comme un outil utile pour la modélisation et l'analyse des réseaux moléculaires. Introduits en 1962 par le Dr Carl Adam Petri comme un formalisme pour la modélisation de la concurrence, ce formalisme a été utilisé dans de nombreuses applications [Petri, 1980, Baldan et al., 2010, Koch et al., 2017] où les différents éléments représentés sont en concurrence.

Les réseaux de Petri sont des structures mathématiques comportant trois types de composants : les Places (notées p_x ; et généralement représentées par des cercles), les Transitions (notées t ; et généralement représentées par des rectangles) et les arcs (représentées par des flèches). Le déplacement des jetons de places en places décrit des transitions d'état qui permettent d'étudier le comportement dynamique du système. Un réseau de Petri évolue lorsqu'on exécute une transition : des jetons sont retirés dans les places en entrée de cette transition et des jetons sont déposés dans

les places en sortie de cette transition. L'exécution d'un réseau de Petri n'est pas déterministe, car il peut y avoir plusieurs possibilités d'évolution à un instant donné (eg. pour une place en amont de deux transitions concurrentes). La distribution des jetons dans les places est appelée le marquage du réseau de Petri.

Plus formellement un réseau de Petri est défini comme suit :

Définition 2.3.1 (Réseau de Petri). Un réseau de Petri est un quintuplet (P, T, I, O, M_0) où :

- P est l'ensemble des places, $P = \{p_0, p_1, \dots, p_i\}$.
- T est l'ensemble des transitions, $T = \{t_0, t_1, \dots, t_j\}$
- I est l'ensemble des arcs dirigés de T vers P étiquetés par un entier, $I : P * T \rightarrow N_0$, avec N_0 appartenant à \mathbb{N}^* .
- O est l'ensemble des arcs dirigés de T vers P étiquetés par un entier appartenant à \mathbb{N}^* , $O : T * P \rightarrow N_0$.
- M_0 est l'état initial, ou le marquage initial du réseau, c'est-à-dire le nombre initial de jetons à chaque place P .

On note M_x le marquage de l'état x du réseau. Il représente le nombre de jetons dans les différentes places P .

La Figure 2.3 illustre le 5-uplet d'un réseau de Petri simple.

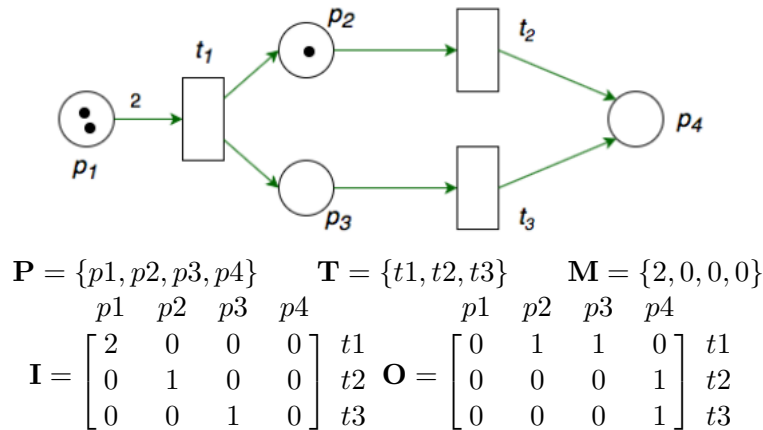


Figure 2.3 – Figure représentant les différents éléments d'un réseau de Petri.

Les états successifs peuvent être calculés à l'aide d'une matrice d'incidence qui contient la différence entre le nombre de jetons produits et le nombre de jetons consommés pour chaque transition. La matrice d'incidence, C est donnée par : $C = O - I$ où O est la matrice de production de transitions et I la matrice de consommation (un exemple table 2.1).

La dynamique d'un réseau de Petri peut être calculée en déterminant successivement tous les états accessibles à partir de l'état initial. Ces états sont ensuite calculables à partir du marquage

$$\begin{array}{cccc}
 p1 & p2 & p3 & p4 \\
 \begin{bmatrix} -2 & 1 & 1 & 0 \\ 0 & -1 & 0 & 1 \\ 0 & 0 & -1 & 1 \end{bmatrix} & t1 & t2 & t3
 \end{array}$$

Table 2.1 – Matrice d’incidence calculée à partir du réseau de Petri décrit figure 2.3.

initial et de la matrice d’incidence.

Ce formalisme permet clairement une représentation fidèle des réactions métaboliques présentes puisque, comme pour une réaction métabolique, le système de places et de transitions capture le phénomène de disparition du substrat au profit de l’apparition du produit. Sur le plan formel, les réseaux de Petri ont les qualités requises pour être le formalisme adéquat. Notons cependant que les réseaux de Petri sont plus spécialisés sur l’analyse des flux comme leurs homonymes EFM, et FBA en version quantitative : la mise en place de tous les éléments pour définir les *réglations* s’avère complexe puisqu’on est *a priori* proche de l’explication mécanistique. Alors que les réseaux de Petri sont dédiés à la gestion de la concurrence sur les ressources, les réglations ne consomment généralement pas leurs ressources, donc il faut ajouter des dispositifs pour « rendre » les ressources consommées lorsqu’on représente un réseau de régulation en Petri [Reddy et al., 1993, Baldan et al., 2010]. La conséquence de ce câblage supplémentaire est qu’il rend plus complexe l’analyse du modèle avec les outils formels.

2.3.2 Formalismes à base de règles de transformation

Les formalismes à base de règles considèrent un stock d’éléments et font évoluer ce stock selon des règles bien définies. Ces formalismes sont donc proches des réactions chimiques, catalysées par les enzymes. A travers ces formalismes on décrit sous forme de règles les réactions d’un réseau. Nous présentons dans cette partie deux formalismes : BIOCHAM [Calzone et al., 2006] et KAPPA [Boutillier et al., 2018].

BIOCHAM BIOCHAM (BIOChemical Abstract Machine) est à la fois un langage à base de règles et un environnement logiciel pour la biologie des systèmes et la biologie systémique. Il fournit des outils d’identification de paramètres et des outils de validation basés sur de nombreux concepts de l’informatique (principalement la logique). Cet environnement permet au modélisateur de définir un ensemble initial de molécules abstraites (appelé solution) ainsi que des règles d’évolution (réactions) à partir desquelles la dynamique peut être déduite. La plate-forme logicielle BIOCHAM offre également la possibilité de déduire de ces modèles à bases de règles d’autres modèles dans des formalismes différents tels que les processus de Markov, ou les EDO, mais nous ne discuterons pas de ces cadres quantitatifs de modélisation ici.

Langage KAPPA Le langage Kappa décrit aussi les réactions biochimiques au niveau moléculaire. La principale différence entre ces deux formalismes est que BIOCHAM est basé sur des règles qui manipulent des termes, alors que les règles de Kappa manipulent des graphes.

Même si dernièrement certaines tentatives de modélisation ont été effectuées au niveau cellulaire avec le langage Kappa [Bouguéon et al., 2021], l’élaboration d’un si grand nombre de réac-

tions reste aujourd'hui très complexe tout comme dans les formalismes quantitatifs présentés dans la section précédente. Il faut donc s'intéresser aux formalismes ayant une forte capacité d'abstraction, permettant par exemple d'abstraire une voie métabolique en une seule entité. L'objectif d'une telle abstraction est bien évidemment de se placer à un niveau cellulaire beaucoup plus intégré, de s'abstraire des détails d'implémentation biologique et de se focaliser sur les interactions entre les grandes voies métaboliques.

2.3.3 Réseaux booléens

Les réseaux booléens ont été introduits par Kaufman puis Thomas [Kauffman, 1969, Thomas, 1973b] pour décrire les systèmes dynamiques de réseau génétique. Simples et puissants, les réseaux booléens se composent de nœuds (représentant les gènes) et des fonctions booléennes qui modélisent la transition entre les différents états du système dynamique (états booléens des différents nœuds). Cet ensemble de fonctions détermine une topologie du réseau qui correspond aux connexions entre les différents gènes. Un exemple de réseau booléen est présenté dans l'introduction du chapitre, figure 2.2, à travers un simple graphe étiqueté où chaque nœud, appelé aussi variable, représente un gène et les arcs abstraient les régulations (« + » pour une activation et « - » pour une inhibition).

Dans ces réseaux lorsque la valeur de la variable est 0 (resp.1), le gène abstrait par cette variable est inactif (resp. actif). Pour un réseau booléen donné avec n nœuds, le système aura 2^n états possibles, chaque état étant modélisé par un vecteur dénotant la valeur booléenne de chaque nœud. Pour construire un modèle à l'aide d'un réseau booléen, quatre étapes sont nécessaires (cf figure 2.4) :

1. *Construction du graphe d'interaction* : il n'est que la transcription des interactions entre les éléments biologiques (voir 1. dans la figure 2.4). Les éléments biologiques constituent les nœuds du graphe et dans le cas du métabolisme ces nœuds peuvent directement représenter des entités métaboliques comme par exemple le glucose, mais peuvent aussi représenter des éléments abstraits qui représentent plusieurs métabolites voire l'activation ou l'inhibition d'une voie métabolique entière (exemple de la glycolyse).
2. *Définition des fonctions booléennes* : les fonctions booléennes décrivent l'activation de chaque variable, en fonction de l'état des autres variables (voir 2. dans la figure 2.4).
3. *Construction des tableaux de transitions entre états* : ces tableaux montrent l'évolution de chaque variable en fonction de l'état de ses prédécesseurs dans le graphe. Sa construction se base donc sur les fonctions booléennes (voir 3. dans la figure 2.4).
4. *Déduction de la dynamique du réseau* : la dynamique d'un réseau se définit à partir des tableaux de transitions. Cette étape nécessite de choisir un mode de mise à jour. La mise à jour peut se faire (pour ne citer que les stratégies les plus courantes) de manière synchrone, synchrone par blocs, ou asynchrone. La sémantique synchrone, implique que lorsque plusieurs variables peuvent être mises à jour, toutes sont mises à jour simultanément. Ainsi, il n'y a qu'une transition possible par état. Dans la sémantique asynchrone par contre, seul un nœud peut être mis à jour à la fois. Ainsi, pour chaque état si plusieurs transitions sont possibles on aura un comportement non déterministe. Enfin, dans les modèles synchrones par blocs, des sous-ensembles de variables sont choisis pour mettre à jour leur valeurs

simultanément dans un ordre prédéfini.

Parmi ces différents types de modèles, nous n'illustrons ici (voir figure 2.5) que les modèles asynchrones car ils sont plus proches des phénomènes biologiques. Dans la pratique, il est improbable que deux entités biologiques passent leurs seuil d'activité en même temps.

Supposons par exemple que l'on sache que Z est activé uniquement lorsque X et Z sont tous les deux simultanément activés. Cette information permet la définition du tableau représentant les évolutions possibles de la variable Y . Ainsi, la « fonction focale » $F(Z) = 1$ si et seulement si $X = 1$ et $Y = 1$. En suivant cette logique on peut remplir les tableaux pour toutes les fonctions booléennes associées aux variables.

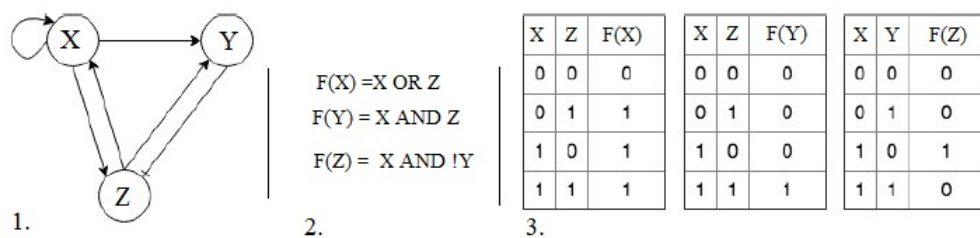


Figure 2.4 – 1) Réseau booléen sur trois variables X, Y et Z. 2) Fonctions booléennes exprimant différents types de coopération. 3) Tables de vérité pour chacun des nœuds.

Pour construire le graphe de transition associé à la figure 2.4, on passe en revue tous les états. Pour chacun d'eux, on détermine pour chaque variable la valeur de sa fonction focale. La fonction focale d'une variable indique si la variable va changer d'état ou non. Lorsque une variable change d'état, elle effectue une transition que l'on va dessiner dans le graphe de transition. Le graphe de transition est donc une représentation du profil d'évolution de toutes les variables état par état. En reprenant notre exemple, si on part de l'état où X et Y sont égaux à 0 et Z à 1, noté (0,0,1), la fonction focale $F(Z)$ nous indique que lorsque $X = Y = 0$, $F(Z) = 0$, $F(Z)$ est différent de la valeur courante de Z donc Z , peut évoluer vers 1. Dans l'état qui succède X et Y restent à 0 et Z devient égal à 0, on le note (0,0,0). Le graphe de transition obtenu est donné en figure 2.5.

L'utilisation des réseaux booléens semble à cette étape très adaptée pour représenter des systèmes de régulation. Cependant, l'expressivité de ces derniers (états booléens) n'est pas à la hauteur de la complexité biologique. En effet, nous voudrions pouvoir définir plusieurs niveaux d'activité (faible, moyen et élevé) pour une même entité biologique et pour cela il nous faut un cadre de modélisation multivalué.

Pour modéliser les régulations du métabolisme, nous souhaitons pouvoir représenter la complexité des vivants à un niveau très abstrait et ne pas se focaliser sur la multitude des réactions biochimiques qui ne sont que le support de régulations de la cellule. Ces exigences nous ont permis de mettre de côté les cadres de modélisation continue puisqu'ils représentent souvent une connaissance mécanistique des régulations.

Parmi les formalismes discrets, les réseaux booléens permettent une description très abstraites des régulations, et sont totalement adaptés à des traitements automatiques pouvant mener à une approche de modélisation assistée très efficacement par ordinateur. Cependant, comme chaque variable ne peut prendre que deux valeurs, ce formalisme souffre d'une trop faible expressivité lors-

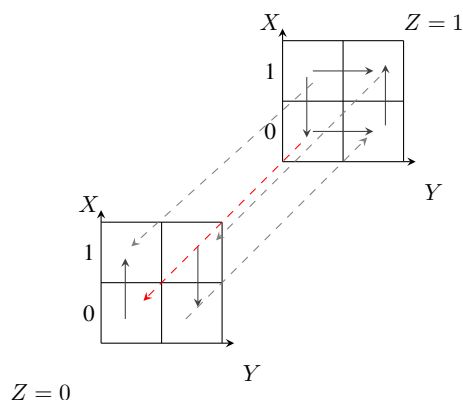


Figure 2.5 – Graphe de transitions des états pour le réseau booléen donné figure 2.4. En rouge la transition de l'état $(0,0,1)$ à l'état $(0,0,0)$

qu'une entité biologique se comporte différemment suivant qu'elle soit présente dans une concentration faible, médiane ou élevée. Nous nous sommes donc tournés vers un formalisme discret fondé sur des concepts identiques (logique), mais qui permet à certaines variables d'avoir plus de 2 niveaux : le formalisme de René Thomas. C'est un formalisme d'abord développé dans un cadre booléen [Thomas, 1973b], pour évoluer dans un cadre multivalué. La cohérence entre les modèles discrets de René Thomas et des modèles à base d'équation différentielles linéaires par morceaux a été démontrée par Snoussi [Snoussi and Thomas, 1993]. Ce cadre de modélisation possède une plateforme d'aide à la modélisation, développé dans notre équipe, appelé *TotemBioNet* dont on discutera chapitre 6.

CHAPITRE 3

Formalisme de René Thomas étendu

*Ce chapitre présente le formalisme de René Thomas pour la modélisation des réseaux de régulation. Les différentes notions et étapes sont illustrées sur un modèle simplifié de la régulation génétique de la production de mucus chez la bactérie *Pseudomonas aeruginosa*. Après une rapide présentation des graphes de régulations, le chapitre met l'accent sur la construction des différentes dynamiques que l'on peut associer à ces graphes. Ces dynamiques sont des graphes de transitions qui dépendent de paramètres. Nous montrons de plus que la logique temporelle CTL et le model checking peuvent aider le modélisateur à choisir les paramétrisations qui sont cohérentes avec les connaissances biologiques sur le comportement du système étudié.*

3.1	Graphe de régulation et paramètres	22
3.1.1	Graphe de régulation avec multiplexes	22
3.1.2	Paramètres	24
3.1.3	Réseau de régulation	25
3.2	Dynamique d'un réseau de régulation	25
3.2.1	Ensemble des états et ressources	25
3.2.2	Paramétrisation	28
3.2.3	Graphe de transitions	29
3.2.4	Boucles de rétroactions	31
3.3	Analyse des traces par une logique temporelle	33
3.3.1	CTL	33
3.3.2	Vérification du réseau de régulation	35
3.3.3	Matrices de phénotypes	37

La modélisation logique définie par R. Thomas [Thomas, 1973b, Thomas, 1978] est utilisée pour décrire et analyser les réseaux de régulation biologique. Enrichi depuis par des méthodes formelles, ce formalisme constitue un très bon cadre pour élaborer des modèles dynamiques qui permettent l'étude théorique des régulations dans un réseau biologique. Ce formalisme discret repose sur la construction d'un Réseau de Régulation (\mathcal{N}), composé d'un graphe de régulation et de paramètres, voir Section 3.1. Les paramètres associés au Graphe de Régulation déterminent ensuite la dynamique du réseau qui est décrite par un graphe de transitions asynchrone *GTA*. Les règles permettant la construction de la dynamique associée sont explicitées en section 3.2. Enfin, pour vérifier que les valeurs des paramètres sont cohérentes avec les connaissances dynamiques connues du système à étudier, on formalise ces dernières à l'aide d'une logique temporelle. Ces comportements serviront pour la vérification formelle, par model Checking, de la cohérence entre comportements connus et dynamique du modèle. Cette phase de vérification pourra être assistée par *TotemBioNet* une plateforme dédiée (voir le chapitre 7). Dans l'ensemble de ce chapitre, on illustrera les concepts en prenant comme exemple jouet le système génétique de production de mucus par la bactérie *Pseudomonas aeruginosa*.

Contexte biologique de la production de mucus chez *Pseudomonas aeruginosa* *Pseudomonas aeruginosa* est une bactérie opportuniste qui peut sécréter du mucus lorsqu'elle est virulente [Pritt et al., 2007, Malhotra et al., 2019]. L'accumulation du mucus produit par *Pseudomonas aeruginosa* peut conduire à la création de biofilms bactériens. Lorsque ces bactéries se logent dans le poumon, elles provoquent des infections graves, en particulier pour les patients atteints de mucoviscidose. Pour ces patients, le mucus produit par *Pseudomonas aeruginosa* crée une couche étanche sur la surface des cellules assimilant l'oxygène, réduisant drastiquement les capacités d'absorption. De manière générale la bactérie peut être sous forme virulente (mucoïde) et produire du mucus, ou non virulente (non-mucoïde). Certains facteurs dans l'environnement de la bactérie influencent l'activation de la mucoïdie [Terry et al., 1991, Martin et al., 1993]. De plus, plusieurs recherches mettent en avant la possibilité que la présence d'ions calcium joue un rôle dans l'activation de la virulence dans un grand nombre de bactéries dont *Pseudomonas aeruginosa* [Sarkisova et al., 2005, Broder et al., 2016, King et al., 2020]

3.1 Graphe de régulation et paramètres

Avant de s'intéresser à la dynamique, il est primordial de bien capturer l'ensemble des interactions mises en jeu à travers un graphe de régulation.

3.1.1 Graphe de régulation avec multiplexes

Le Graphe de Régulation, ou *GR*, constitue la représentation statique des régulations connues du système biologique étudié. Le *GR* est composé de trois éléments :

V est l'ensemble des *variables* mises en interaction. Elles sont représentées sous forme de cercles (voir figure 3.1). La couleur pourra varier en fonction du sens biologique donné à la variable. Leur nom est généralement donné par les éléments biologiques qu'elles représentent. Le formalisme permettant l'abstraction, il est possible de représenter par une variable un ensemble d'éléments biologiques. Par exemple, dans le cadre du métabolisme,

elles peuvent abstraire aussi bien des métabolites que voies métaboliques. Nous reviendrons plus en détail sur cette cohabitation de variables concrètes (métabolites) et variables abstraites (voie métabolique) au chapitre 9 lors de la description du modèle de la régulation du métabolisme cellulaire.

M est l'ensemble des *Multiplexes* qui représentent des règles de régulation entre les variables du système. Par exemple lorsque deux variables A et B sont nécessaires ensemble pour activer une troisième variable C , cette régulation sera représentée par un multiplexe appelé “coopération”, qui explicitera que pour réguler C il faut $(A \geq 1) \& (B \geq 1)$. Chaque multiplexe est représenté graphiquement par un rectangle découpé en deux parties : dans la partie supérieure est indiqué le nom du multiplexe et dans la partie inférieure est indiquée la règle de régulation exprimée par d'une formule logique (figure 3.1).

A est l'ensemble des arcs qui forment les liens de régulation. Chaque arc part d'un multiplexe et va vers une variable. Cela correspond aux flèches en trait plein dans le graphe de régulation (cf figure 3.1). Pour aider la lecture d'un graphe de régulation, nous matérialiserons la participation d'une variable à un multiplexe par une flèche en pointillés. Formellement ces flèches ne sont pas nécessaires car la simple présence d'une variable dans la formule d'un multiplexe exprime que cette variable joue un rôle dans la régulation associée au multiplexe.

Définition 3.1.1 (Graphe de régulation avec multiplexes). *Un graphe de régulation $GR = (V, M, A)$ est un graphe orienté tel que :*

- v et M sont les ensembles de nœuds du graphe tels que $V \cap M = \emptyset$ et dont les éléments sont appelés respectivement variables et multiplexes,
- A chaque variable $v \in V$ est associée un domaine discret $\llbracket 0, b_v \rrbracket$ où $b_v \in \mathbb{N}^*$,
- Les arcs de A partent chacun d'un multiplexe et vont vers une variable ($A \subset M \times V$),
- Chaque multiplexe $m \in M$ a une formule associée φ_m qui exprime la condition sous laquelle m influence sa (ses) variable(s) cibles. Le langage des formules des multiplexes est défini inductivement par :
 - Les atomes sont les formules de la forme $(v \geq n)$ avec $v \in V$ et $n \in \llbracket 0, b_v \rrbracket$
 - si φ, φ_1 et φ_2 sont des formules de multiplexes, alors $\neg\varphi, \varphi_1 \square \varphi_2$ sont aussi des formules de multiplexes, où $\square \in \{\wedge, \vee, \Rightarrow\}$.

*Exemple 3.1.1 – Graphe de régulation de la production de mucus chez *Pseudomonas aeruginosa*.*

Le mucus est dû à la présence de la protéine alginate, $AlgU$. $MucB$ est une autre protéine, exprimée en même temps qu' $AlgU$ par l'intermédiaire de l'activation d'un élément génétique appelé *opéron*. $MucB$ est une protéine de signalisation localisée dans la membrane interne de la bactérie. Lorsqu'il y a accumulation d'alginate dans l'environnement cellulaire, la protéine $MucB$ capte cette accumulation et transmet un signal qui permet indirectement l'inhibition de l'opéron. L'opéron, quant à lui, est capable d'auto-activation via les molécules intermédiaires abstraites dans le modèle. Finalement, on considérera que le calcium favorise la transition vers le phénotype virulent de *Pseudomonas aeruginosa* par l'activation de l'opéron. L'ensemble de ces influences individuelles est résumé dans le graphe de régulation figure 3.1.

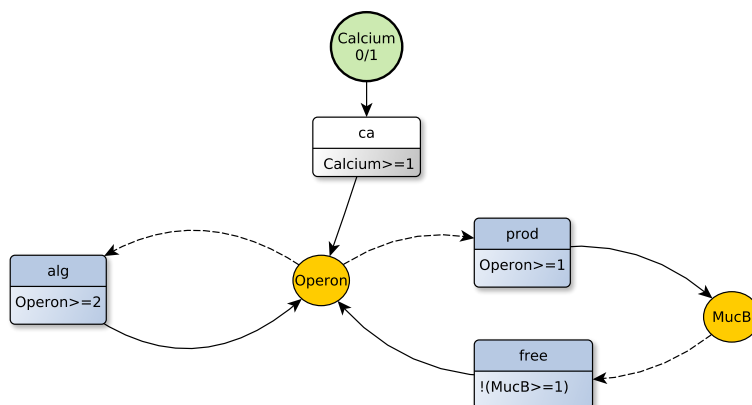


Figure 3.1 – Graphe de régulation pour le système de production de mucus chez *Pseudomonas aeruginosa*. Les variables sont représentées par des cercles et les multiplexes par des rectangles.

Ce graphe de régulation détermine un ensemble de paramètres dont les valeurs piloteront la dynamique du modèle.

3.1.2 Paramètres

Par convention, les *paramètres* d'une variable v sont notés $K_{v,\omega}$ où ω est un sous-ensemble des prédécesseurs de v dans GR . $K_{v,\omega}$ représente la valeur vers laquelle la variable v tend lorsque les seuls multiplexes prédécesseurs dont la formule est évaluée à vraie sont ceux de ω . Avec cette notation, les paramètres montrés graphiquement dans la figure 3.2 sont respectivement notés $K_{MucB, \{\}} et $K_{MucB, \{\text{prod}\}}$. Notons que pour la syntaxe concrète, dans la suite du manuscrit, nous ne mettrons ni, accolades ni virgules pour écrire l'ensemble ω (on écrit donc comme un mot sur l'alphabet des prédécesseurs de v).$

Exemple 3.1.2 – Pour *Pseudomonas aeruginosa*, la variable $MucB$ a 2 paramètres (voir figure 3.2). K_{MucB} (à gauche) est celui sans ressource de régulation (la formule du multiplexe **Prod** est évaluée à faux) alors que $K_{MucB, \text{prod}}$ (à droite) est celui où la régulation par le prédécesseur $Prod$ est active (la formule du multiplexe est évaluée à vrai). Le nombre de paramètres d'une variable correspond donc au nombre des parties de l'ensemble de ses prédécesseurs : $2^{d^-(v)}$, où $d^-(v)$ est le nombre de prédécesseurs de v dans GR . On note \mathcal{K} la famille des $K_{v,\omega}$

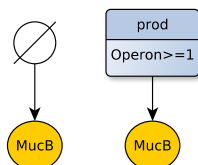


Figure 3.2 – Deux configurations possibles de prédécesseurs pour $MucB$.

3.1.3 Réseau de régulation

Le couple graphe de régulation et paramètres forme un *Réseau de Régulation*, $\mathcal{N} = (GR, \mathcal{K})$, visualisable dans un tableau.

Exemple 3.1.3 – L’ensemble des éléments du réseau de production du mucus chez *Pseudomonas aeruginosa* est listé dans le tableau suivant :

GR			K		
V	M	A			
Calcium	<i>ca</i>	<i>ca</i> → Operon	$K_{Operon, ca}$	K_{Operon}	K_{MucB}
MucB	<i>alg</i>	<i>alg</i> → Operon	$K_{Operon, free ca}$	$K_{Operon, free}$	$K_{MucB, prod}$
Operon	<i>prod</i>	<i>prod</i> → MucB	$K_{Operon, alg ca}$	$K_{Operon, alg}$	$K_{calcium}$
	<i>free</i>	<i>free</i> → Operon	$K_{Operon, alg free ca}$	$K_{Operon, alg free}$	

Table 3.1 – Tableau représentant le réseau de régulation de la production de mucus chez *Pseudomonas aeruginosa*.

Chaque paramètre $K_{v, \omega}$ prendra une valeur du domaine $\llbracket 0, b_v \rrbracket$ de la variable v associée. Ce sont les valeurs associées aux différents paramètres qui permettront de définir la dynamique du système. L’identification des valeurs des paramètres est donc primordiale.

3.2 Dynamique d’un réseau de régulation

Pour construire les dynamique d’un système, commençons par définir les notions d’état et d’ensemble d’états. On pourra ensuite définir ce qu’est une ressource, et déterminer les paramètres à appliquer en fonction des ressources de l’état courant. Finalement, on pourra construire un Graphe de Transitions Asynchrone (GTA) représentant la dynamique associée à la paramétrisation choisie du réseau de régulation.

3.2.1 Ensemble des états et ressources

Un état d’un système est défini par l’ensemble des valeurs prises par chacune des variables du système.

Définition 3.2.1 (les états). L’ensemble des états d’un système est le produit cartésien $S = \prod_{v \in V} \llbracket 0, b_v \rrbracket$. On note $s \in S$ un état.

Exemple 3.2.1 – Pour *Pseudomonas aeruginosa* le nombre total d’états possibles correspond au produit du nombre de valeurs que chaque variable peut prendre : $3 \times 2 \times 2 = 12$ car les domaines de **Operon MucB** et **calcium** sont respectivement, $\llbracket 0, 2 \rrbracket$, $\llbracket 0, 1 \rrbracket$ et $\llbracket 0, 1 \rrbracket$. Tous ces états sont représentés dans la figure 3.3. Pour représenter la dynamique de ce système, il faut représenter chacun

de ces 12 états. On pourrait les représenter par des cubes, dans un espace 3D, mais ici pour simplifier nous les avons dessinés par des carrés dans deux plans. La Troisième dimension est donc présente par l'existence de deux plans avec à gauche le cas où **Calcium**=0 et à droite le cas où **Calcium**=1.

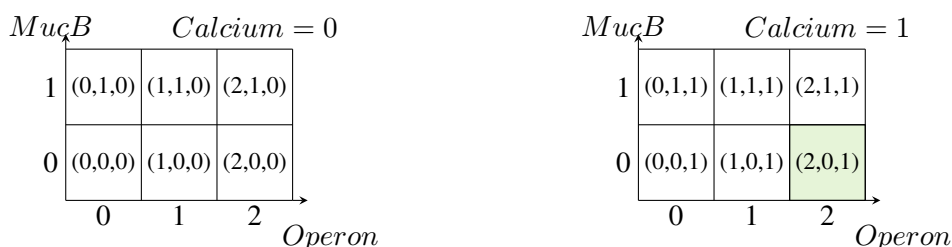


Figure 3.3 – Graphe des états pour *Pseudomonas aeruginosa*.

Considérons l'état courant $(2,0,1)$, représenté sur fond vert dans le graphe des états et focalisons-nous sur la variable **Operon**. Si l'on se réfère au graphe de régulation de la figure 3.1 les prédécesseurs possibles de **Operon** sont : **ca. prod** et **free**. voir Fig. 3.4.

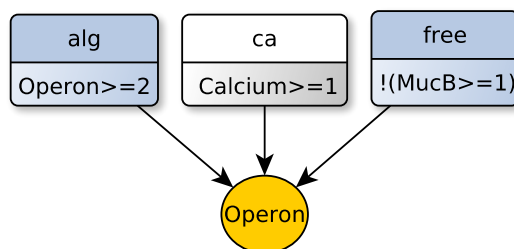


Figure 3.4 – Schéma de la variable opérone et de l'ensemble de ses prédécesseurs

La régulation due au multiplexe **ca** est active dans l'état courant puisque la condition de la formule du multiplexe ($\text{Calcium} \geq 1$) est vérifiée. Il en est de même pour le multiplexe **alg**. Toujours dans l'état courant (où **MucB** prend la valeur 0) la formule du multiplexe **free** est aussi satisfaite car $MucB = 0$. On introduit ainsi la définition de ressource d'une variable v dans un état s : un prédécesseur de la variable v sera une ressource si la formule associée est vraie dans l'état s .

Définition 3.2.2 (Ressources). *Chaque multiplexe m , prédécesseur de v , tel que φ_m est satisfait dans un état s est appelé une ressource de v dans s . Lorsqu'il n'y a pas d'ambiguïté sur la variable v par le contexte, on notera ω_s l'ensemble des ressources de v dans l'état s .*

La formule du multiplexe **free** indique donc qu'il y a ressource lorsque il y a absence de **MucB** : il s'agit d'une inhibition. Notons que la présence (resp. absence) d'un activateur, aura pour conséquence la présence (resp. absence) de la ressource correspondante. A l'inverse lorsque l'on étudie une inhibition, lorsque la ressource est présente (la formule du multiplexe est satisfaite), l'inhibiteur est absent. L'absence de l'inhibition *via free* revient donc à une activation. Ainsi, on considère l'absence d'un régulateur négatif comme une activation dans ce formalisme.

Cette remarque permet de mettre en avant une propriété importante dans ce formalisme : comme toute ressource représente une activation, cela signifie que plus on a de ressources plus la variable régulée peut être activée. On en déduit les conditions de Snoussi [Snoussi and Thomas, 1993] :

Définition 3.2.3 (Conditions de Snoussi). Soit $K_{v,\omega}$ et $K_{v,\omega \cup \{x\}}$ deux paramètres de la variable v où ω est un ensemble de ressources de v et x une ressource de v telle que $x \notin \omega$ alors $K_{v,\omega \cup \{x\}} \geq K_{v,\omega}$.

A l'aide des ressources nous pouvons maintenant définir les paramètres applicables dans un état particulier.

Définition 3.2.4 (Paramètre applicable dans un état). Étant donné un GR, un état $s \in S$ et une variable $v \in V$, le paramètre applicable de v dans s , est l'unique paramètre $K_{v,\omega}$ tel que,

- $\forall m \in \omega$, φ_m est vraie dans s ,
- $\forall m \in \text{pred}(v)$ tel que $m \notin \omega$, φ_m est faux dans s .

Le paramètre applicable de v dans l'état s est noté K_{v,ω_s} .

Exemple 3.2.2 – Le tableau ci-dessous liste l'ensemble des états possibles et associe à chacun de ces états les paramètres applicables pour chacune des variables. La ligne soulignée en vert correspond à l'état (2,0,1). Les paramètres sur fond vert sont ceux qui seront applicables pour cet état.

États s			Paramètres applicables		
s(Operon)	s(MucB)	s(calcium)	K_{MucB,ω_s}	K_{Operon,ω_s}	$K_{Calcium,\omega_s}$
0	0	0	K_{MucB}	$K_{Operon, \mathbf{free}}$	$K_{Calcium}$
1	0	0	$K_{MucB, \mathbf{prod}}$	$K_{Operon, \mathbf{free}}$	$K_{Calcium}$
2	0	0	$K_{MucB, \mathbf{prod}}$	$K_{Operon, \mathbf{alg free}}$	$K_{Calcium}$
0	1	0	K_{MucB}	K_{Operon}	$K_{Calcium}$
1	1	0	$K_{MucB, \mathbf{prod}}$	K_{Operon}	$K_{Calcium}$
2	1	0	$K_{MucB, \mathbf{prod}}$	$K_{Operon, \mathbf{alg}}$	$K_{Calcium}$
0	0	1	K_{MucB}	$K_{Operon, \mathbf{free ca}}$	$K_{Calcium}$
1	0	1	$K_{MucB, \mathbf{prod}}$	$K_{Operon, \mathbf{free ca}}$	$K_{Calcium}$
2	0	1	$K_{MucB, \mathbf{prod}}$	$K_{Operon, \mathbf{alg free ca}}$	$K_{Calcium}$
0	1	1	K_{MucB}	$K_{Operon, \mathbf{ca}}$	$K_{Calcium}$
1	1	1	$K_{MucB, \mathbf{prod}}$	$K_{Operon, \mathbf{ca}}$	$K_{Calcium}$
2	1	1	$K_{MucB, \mathbf{prod}}$	$K_{Operon, \mathbf{alg ca}}$	$K_{Calcium}$

Table 3.2 – Tableau récapitulatif des paramètres applicables pour *Pseudomonas aeruginosa*. Lorsque **MucB** est absent alors **Operon** a **free** dans ses ressources car c'est un inhibiteur. L'absence d'un inhibiteur joue un rôle similaire à la présence d'un activateur et inversement. La ligne soulignée en vert correspond à l'état (2,0,1) utilisé comme exemple.

Il existe des cas où certains paramètres ne sont applicables dans aucun des états à cause de la structure même du graphe de régulation. Cela amène à la notion de paramètre opérable.

Définition 3.2.5 (Paramètres opérables). *Le paramètre $K_{v,\omega}$ est dit opérable s'il existe au moins un état s dans lequel $K_{v,\omega}$ est applicable.*

Remarquez qu'un paramètre *non opérable* est inutile dans la construction du graphe de transitions d'états asynchrones.

Exemple 3.2.3 – Supposons une variable v sous le contrôle de deux multiplexes μ_1 et μ_2 dont les formules sont respectivement $\varphi_1 \equiv (a \geq 2)$ et $\varphi_2 \equiv (a < 2)$. Le paramètre $K_{v,\mu_1 \ \mu_2}$ est non opérable puisqu'il n'existe aucun état où les deux formules de multiplexes sont simultanément vraies.

3.2.2 Paramétrisation

Pour pouvoir associer une dynamique au graphe de régulation, on va donner une valeur à chaque paramètre opérable. L'ensemble des paramètres valués constituera une paramétrisation. Le nombre de paramétrisations est majoré par :

$$\prod_{v \in V} (b_v + 1)^{(2^{i_v})} \quad (3.1)$$

avec i_v le degré entrant de la variable v (nombre de régulations) et b_v la borne de v . La condition de Snoussi et l'existence de paramètres non opérables peut diminuer cette valeur mais n'en change pas l'ordre de grandeur en général. Nous présenterons dans le chapitre 7 différentes méthodes permettant d'identifier les paramètres associés à un graphe de régulation.

Définition 3.2.6 (Paramétrisation). *Étant donné un réseau de régulation $\mathcal{N} = (GR, \mathcal{K})$, une paramétrisation P est une substitution :*

$$P: \mathcal{K} \rightarrow \mathbb{N}$$

telle que $P(K_{v,\omega}) \in \llbracket 0, b_v \rrbracket$.

Intuitivement, la valeur de chaque paramètre applicable $K_{v,\omega}$ dans un état s constitue le point focal (\mathcal{F}) de la variable v dans s , c'est à dire la valeur vers laquelle la variable v est attirée. Cette valeur est évidemment comprise entre 0 et b_v .

Définition 3.2.7 (Point focal d'un état). *Le point focal \mathcal{F} d'un état s est l'état dans lequel les composantes de toutes les variables ont comme valeur la valeur focale à l'état s , autrement dit, l'état s' tel que, pour tout $v \in V$, $s'(v) = K_{v,\mathcal{F}(s,v)}$*

Exemple 3.2.4 – La table 3.3 indique une paramétrisation possible parmi d’autres pour les paramètres du réseau de régulation associé à *Pseudomonas aeruginosa*.

$K_{Operon, \mathbf{ca}} = 0$	$K_{Operon} = 0$
$K_{Operon, \mathbf{free\ ca}} = 1$	$K_{Operon, \mathbf{free}} = 0$
$K_{Operon, \mathbf{alg\ ca}} = 1$	$K_{Operon, \mathbf{alg}} = 1$
$K_{Operon, \mathbf{alg, free, ca}} = 2$	$K_{Operon, \mathbf{alg\ free}} = 2$
$K_{MucB} = 0$	$K_{MucB, \mathbf{prod}} = 1$
$K_{Calcium} = 1$	

Table 3.3 – Exemple d’une paramétrisation pour *Pseudomonas aeruginosa*. Le tableau contient l’ensemble des paramètres et points focaux associés. En vert sont représentés les paramètres qui s’appliquent dans l’état (2,0,1) utilisé comme exemple.

3.2.3 Graphe de transitions

A ce stade nous avons défini le graphe de régulation et pour chaque état quels les paramètres vont s’appliquer. Cependant, le système n’ira pas forcément de l’état courant à l’état focal : en effet, ce point focal ne définit que la direction locale d’évolution du système. La pertinence du cadre de modélisation de R.Thomas repose sur deux idées maîtresses pour représenter l’évolution d’un système biologique de régulation : la désynchronisation et la transition unitaire.

Désynchronisation Supposons qu’au moins deux variables sont attirées vers des valeurs différentes de leur état courant. Rappelons d’abord que les niveaux des variables abstraites représentent soit des activités soit des concentrations. Dans l’exemple *Pseudomonas aeruginosa* *MucB* et *Operon* représentent respectivement une concentration protéique et une activité. La probabilité que deux variables franchissent *simultanément* leur seuil respectif (exactement au même moment) est nulle¹. Néanmoins, nous ne pouvons pas savoir quel seuil sera traversé en premier. En effet, un état s signifie que le niveau qualifie que le niveau quantitatif de la variable se situe dans le n -ième intervalle mais sans précisions de sa distance des bornes de cet intervalle. De plus, la biologie met en jeu un fort indéterminisme (sur les productions/ dégradations de molécules en particulier). Il faut donc modéliser ce non déterminisme. Pour cela on ne considérera qu’un seul passage de seuil à la fois pour chaque transition d’un état à un autre et le non déterminisme porte sur le choix de la variable qui passera son seuil en premier.

Transitions unitaires Considérons maintenant le cas où la valeur focale d’une variable, dans un état donné, traverse plusieurs états pour l’atteindre par exemple passer de s à $s + 2$. Cependant, l’augmentation ou la diminution de l’activité ou de la concentration d’une variable ne pourra se faire que progressivement : la variable ne pourra pas « sauter » directement de l’état courant à la valeur focale. La variable passera d’abord dans le niveau voisin du niveau courant ($s + 1$). Chacune des transitions est donc de longueur 1 dans la direction de la valeur focale. Si le paramètre indique une valeur focale plus grande (resp. plus petite) que la valeur courante, il sera possible d’augmenter

1. Notons que le seuil n n’est pas forcément placé au même endroit pour toutes les variables. Cette particularité laisse une grande marge de manœuvre pour représenter les systèmes biologiques.

(resp. diminuer) le niveau de la variable, mais il ne sera pas possible de faire un “bond” pour enjamber plusieurs seuils en une seule étape.

On peut maintenant définir le graphe de transitions asynchrone.

Définition 3.2.8 (*GTA pour une paramétrisation P*). Étant donné un réseau de régulation $\mathcal{N} = (GR, \mathcal{K})$ dont les variables sont désignées par v_1, \dots, v_n et $P \in P_{\mathcal{N}}$ une paramétrisation pour \mathcal{N} , le graphe de transitions asynchrones GTA_P associé à P est défini comme suit :

- **l’ensemble des noeuds** : est l’ensemble des états,
- **l’ensemble des Arcs** : il existe un arc de $s^p = (s_1^p, \dots, s_n^p)$ à $s^q = (s_1^q, \dots, s_n^q)$ s’il existe un et un seul indice i tel que, $s_i^p \neq s_i^q$ avec soit : $s_i^q = s_i^p + 1$ et $P(K_{v_i, \omega_{s^p}}) > s_i^p$ soit $s_i^q = s_i^p - 1$ et $P(K_{v_i, \omega_{s^p}}) < s_i^p$.

Exemple 3.2.5 – Pour illustrer les étapes de création de ce graphe nous allons nous référer aux tableaux 3.2 et 3.3. Pour chaque état, le tableau 3.2 donne les paramètres applicables, puis on regarde dans le second tableau 3.3 leurs valeurs. Le graphe de transitions asynchrone peut alors être construit transition par transition. Lorsque l’un des paramètres applicables dans un état s donné a une valeur différente de l’état de la variable v associée, alors il existe une transition qui part de l’état considéré vers l’état voisin où seule la valeur de la variable v a augmenté de 1 ($K_{v, \omega_s} \geq s_v$) ou a diminué de 1 ($K_{v, \omega_s} \leq s_v$).

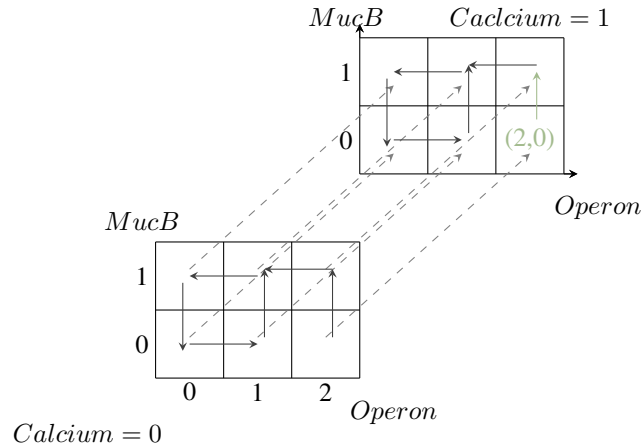


Figure 3.5 – graphe de transitions pour *Pseudomonas aeruginosa* après la table 3.3.

Si on reprend l’état : $(2,0,1)$ dans le tableau 3.2, on constate que les trois paramètres qui s’appliquent sont $K_{Calcium}$, $K_{MucB, \text{prod}}$ et $K_{Operon, \text{free ca}}$. Leurs valeurs focales respectives sont 2, 1 et 1 (tableau 3.3). On constate que la variable **MucB** passe de 0 à 1. On dessine donc dans le graphe de transitions une flèche de l’état $(2,0,1)$ vers $(2,1,1)$. Comme les valeurs des paramètres applicables associées aux variables **Calcium** et **Operon** ne sont pas différentes leur valeur dans l’état, il n’y a pas d’autres transitions à dessiner.

Cet exemple était un cas simple puisque une seule variable pouvait évoluer. Si plusieurs variables doivent changer d’états on doit faire autant de flèches différentes qu’il y a de variables qui

peuvent évoluer (principe de désynchronisation). De même, en suivant le principe de transition unitaire, aucune flèche n’est autorisée à traverser un état sans s’y arrêter : il n’existe donc pas de flèche de taille supérieure à 1. Après avoir suivi cette méthode pour l’ensemble des états du \mathcal{N} on obtient le graphe de transitions Fig 3.5.

Nous avons maintenant terminé les définitions pour la formalisation des modèles. Sur cette base on peut étudier le graphe de régulation, le graphe de transitions et faire des liens entre les deux graphes. D’après les conjectures de René Thomas [Snoussi and Thomas, 1993] prouvées depuis, ce sont les boucles de rétroactions positives et négatives qui pilotent les différents comportements globaux du système de régulation étudié.

3.2.4 Boucles de rétroactions

Si l’on se penche sur l’étude du graphe de régulation de *Pseudomonas aeruginosa*, on constate qu’il existe deux boucles de rétroactions : l’auto-régulation de l’opéron sur lui même, et la boucle de rétroaction entre *Operon* et *MucB*. On dira qu’une boucle de rétroaction est positive (resp. négative) si l’augmentation d’une des variables mène à une action directe ou indirecte positive (resp. négative) sur elle même. Ainsi l’auto-régulation de l’opéron constitue une boucle positive et l’autre boucle est négative (voir, Figure 3.1). Étudions les caractéristiques comportementales de ces deux types de boucles.

Les boucles positives. Un cycle est positif s’il contient un nombre pair d’actions négatives (inhibitions).

Exemple 3.2.6 – Deux exemples sont présentés figure 3.6. Dans le schéma de gauche, il y a deux actions positives : x active y et y active x , dans le schéma de droite, il y a deux actions négatives : x inhibe y et y active x . Ces deux exemples constituent des boucles positives.

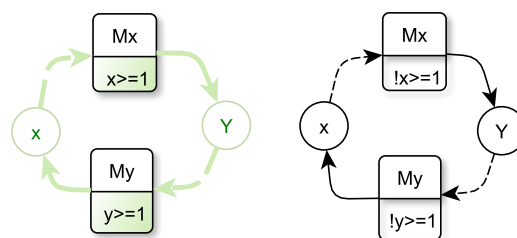


Figure 3.6 – Deux exemples de cycles positifs.

En choisissant une paramétrisation adéquate, on peut rendre de telles des boucles « fonctionnelles ». Notons que x et y sont toutes les deux des variables booléennes pour les deux graphes de régulations proposés. La figure 3.7 donne les deux graphes de transitions respectifs associés aux graphes de régulation donnés figure 3.6 en prenant en compte la paramétrisation suivante dans les deux cas : $K_x = 0$, $K_y = 0$, $K_{x, My} = 1$ et $K_{y, Mx} = 1$. Les deux graphes de transitions exhibent des états dans lesquels le système reste éternellement (états sans arc sortant). On appelle ces états les *états stables*. Plus généralement plusieurs états peuvent ainsi former des bassins d’attractions.

Lorsqu'une boucle positive entraîne l'existence de deux comportements asymptotiques différents alors la boucle est fonctionnelle. Dans le cas général, une boucle positive fonctionnelle peut mener à bassins d'attractions c'est le cas dans nos exemples : on a alors une *multi-stationnarité*.



Figure 3.7 – Graphes de transitions possibles associés aux graphes de régulation de la figure 3.6 (cycle positifs). En vert (resp. noir) le graphe de transitions associé au graphe de régulation vert (resp. noir).

Les boucles négatives. Un cycle négatif est un cycle dans lequel il y a un nombre impair d'actions négatives.

Exemple 3.2.7 – Dans la figure 3.8 sont présentées deux boucles négatives. A gauche x active y et y inhibe x (en vert). A droite x inhibe y et y active x (noir). Les deux variables x et y sont booléennes pour les deux graphes de régulation proposés.

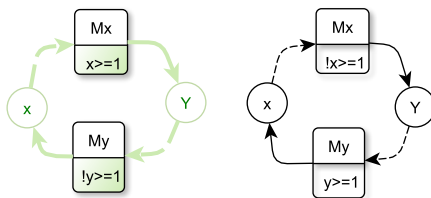


Figure 3.8 – Deux exemple de cycles négatifs.

La figure 3.9 représente les graphes de transitions respectifs des deux graphes de régulations de la figure 3.8 en considérant la paramétrisation suivante dans les deux cas : $K_x = 0, K_y = 0, K_{x, \mathbf{My}} = 1$ et $K_{y, \mathbf{Mx}} = 1$. Nous observons que les transitions forment un cycle, qui représente des oscillations. Lorsqu'on observera une oscillation on sera en présence d'une boucle négative fonctionnelle.



Figure 3.9 – Graphes de transitions associés aux graphes de régulation de la figure 3.8 (cycle négatifs).

La construction d'un modèle, c'est-à-dire d'un graphe de régulation avec ses paramètres, n'est pas suffisante pour avoir un modèle valide. Il faut s'assurer que ce modèle représente bien le système à modéliser. Pour cela on va formaliser les comportements globaux (phénotypes) du système biologique observés par les biologistes et vérifier que ces propriétés sont aussi vérifiées par le modèle proposé. Cette vérification sera assurée par du model checking. La formalisation de ces comportements attendus se fait à l'aide d'une logique temporelle particulière : *CTL* pour *computation tree logic*.

3.3 Analyse des traces par une logique temporelle

Dans ce document nous allons utiliser une logique temporelle propositionnelle, c'est à dire, basée sur un ensemble de propositions atomiques. Ces propositions atomiques sont combinées grâce à un certain nombre de connecteurs, dont les connecteurs logiques classiques : $\wedge, \vee, \Rightarrow$. On y ajoute d'autres opérateurs temporels que l'on appelle les *modalités*. Les logiques temporelles font parties des logiques dites modales.

A l'aide de ce type de logique, il devient possible de caractériser des propriétés comportementales et de confronter ces propriétés aux chemins du graphe de transitions. Si les traces du graphe de transitions satisfont les propriétés comportementales spécifiées, alors le modèle sera validé puisqu'il partage les propriétés du système biologique à modéliser.

Dans cette section nous introduirons d'abord la logique *CTL* puis la logique appelée *Fair-Path CTL*, spécifiquement intéressante lorsqu'on étudie les comportements asymptotiques.

3.3.1 *CTL*

CTL (Computation Tree logic) a été introduit en 1981 par M. Clarke and E. Allen Emerson [Clarke and Emerson, 1982]. C'est une logique temporelle dans laquelle le temps a une structure arborescente. A partir d'un état initial on représente l'ensemble des traces possible par un arbre : l'état initial est la racine de l'arbre. Puis le processus est itéré à partir de chacun des états suivants, les branches de l'arbre sont rendues infinies en ajoutant une transition de chaque état stable vers lui même. Une formule *CTL* représentera de manière formelle un comportement biologique du système étudié. Ces formules sont construites sur les variables v du réseau de régulation. Pour exprimer des propriétés temporelles, *CTL* fait appel à des modalités sur les chemins : E (resp. A) signifiera qu'il existe un chemin à partir de l'état initial (resp. pour tous les chemins à partir de l'état initial) qui satisfait la propriété. De plus, *CTL* combine ces quantificateurs de chemins avec des quantificateurs le long du chemin : X (neXt, au prochain état), F (Future, un état le long du chemin), G (Generally, à chaque état du chemin), U (Until, jusqu'à ce que), voir [Huth and Ryan, 2000]. La syntaxe *CTL* se définit comme suit :

Définition 3.3.1 (Langage *CTL*). *Étant donné un graphe de régulation GR , le langage \mathcal{L}_V des formules *CTL* sur v est défini inductivement par :*

- *Un atome est soit une constante booléenne (\top, \perp), soit de la forme $v \leq n$ avec $v \in V$, $n \in \llbracket 0, b_v \rrbracket$*

- si φ_1 et φ_2 sont des formules CTL alors :
 - $\neg\varphi_1, \varphi_1 \nabla \varphi_2$ où ∇ est un connecteur logique parmi $\wedge, \vee, \Rightarrow, \Leftrightarrow$
 - $E(\varphi_1 U \varphi_2)$ et $A(\varphi_1 U \varphi_2)$
 - $\Box\varphi_1$ où \Box est dans EX, EF, EG, AX, AF, AG
- sont également des formules CTL.

Donnons maintenant une traduction littérale de ces modalités :

EX... peut se traduire par « il existe au moins un chemin tel que dans le prochain état... » ;

AX... par « dans tous les chemins le prochain état est tel que... » ;

EF... par « il existe au moins un chemin tel que dans un futur état... » ;

AF... par « dans tous les chemins il existe un état futur dans lequel... » ;

EG... par « il existe un chemin le long duquel la propriété... est satisfaite à chaque état » ;

AG... par « quelque soit le chemin choisi, il passera par un état futur qui satisfera... » ;

Puis les deux dernières modalités temporelles sont particulières car elles nécessitent deux sous formules. En effet « U » doit être entourée de connecteurs. Par exemple $\varphi U \varrho$ indique que la propriété φ est satisfaite jusqu'à ce que la propriété ϱ le soit :

E[$\varphi U \varrho$] « il existe au moins un chemin le long duquel φ est constamment satisfaite jusqu'à ce que le chemin rencontre un état où ϱ est satisfaite » ;

A[$\varphi U \varrho$] « quelque soit le chemin choisi φ est constamment satisfaite jusqu'à ce que le chemin rencontre un état où ϱ est satisfaite » ;

La figure 3.10 ci-dessous représente ces modalités décrites sous forme arborescente.

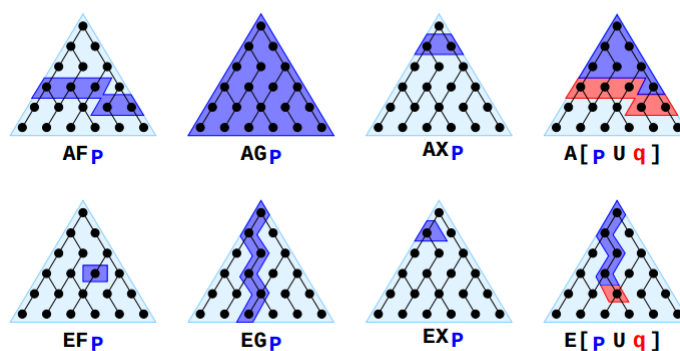


Figure 3.10 – Schéma récapitulatif des différents connecteurs temporels de CTL. Issu du cours en ligne de Antonio Gonzàles Bergueño pour l'université d'Oslo, <https://www.uio.no/studier/emner/matnat/ifi/INF5140/v17/slides/clt-antonio.pdf>.

Reprenons l'exemple de *Pseudomonas aeruginosa*, nous savons qu'une bactérie ne créera jamais de mucus lorsque ses facteurs de virulence ne sont pas activés. Autrement dit, quand on part

d'un état où l'opéron n'est pas activé ($Operon = 0$), on souhaite que, dans le modèle, l'opéron ne soit jamais assez actif pour que le mucus soit produit : $!Operon = 2$ (car l'opéron ne permet la production de mucus que lorsque l' $Operon$ est à son état le plus haut, soit 2). Reste maintenant à décrire le lien temporel qui relie ces deux littéraux.

Dès que l'état initial est tel que $Operon=0$, on sait que l' $Operon$ ne sera jamais actif (au niveau 2). On peut donc traduire ces informations par le connecteur « AG ». La formule représentant cette connaissance biologique s'écrit donc :

$$\varphi^0 \equiv ((Operon = 0) \Rightarrow AG(\neg(Operon = 2)))$$

Ici, la formule paraît relativement simple mais nous verrons par la suite que sur de plus gros modèles, la définition d'un comportement global nécessite un travail minutieux. De plus, malgré sa puissance, ce langage peut parfois induire le modélisateur en erreur car CTL est une logique sans tiers exclus (i.e. φ et $\neg\varphi$ peuvent être toutes les deux fausses).

3.3.2 Vérification du réseau de régulation

Une fois construites, les formules CTL représentant les comportements attendus du modèle peuvent être confrontées au graphe de transitions. Cela permet la vérification de la cohérence de la dynamique du réseau de régulation avec les comportements souhaités, c'est le principe du model-checking.

Notation – Étant donné un graphe de régulation \mathcal{N} , le model-checking $check_{\mathcal{N}}$ pour les formules dans \mathcal{L}_V est noté :

$$check_{\mathcal{N}} : P_{\mathcal{N}} \times \mathcal{L}_v \rightarrow \text{Bool}$$

où $check_{\mathcal{N}}(P, \varphi) = true$ ssi φ est vraie pour le graphe de transitions GTA_P construit pour le \mathcal{N} avec la paramétrisation P à partir de n'importe quel état initial.

Reprenons le graphe de transitions défini précédemment et regardons si la formule CTL $\varphi^0 \equiv ((Operon = 0) \Rightarrow AG(\neg(Operon = 2)))$ est satisfaite (voir figure 3.11). En partant des états où $Operon=0$ (états en fond vert), on constate qu'aucun des chemins ne mène à $Operon=2$ (cases représentées sur fond rouge). La formule CTL est donc vérifiée pour le réseau de régulation proposé. Supposons que cette formule représente la seule information sur le comportement global de *Pseudomonas aeruginosa* alors on considère que le modèle est valide. La dynamique de ce modèle pourra ensuite être étudiée.

Définition 3.3.2 (Modèles de $\varphi \in \mathcal{L}_V$). Étant donné un réseau de régulation \mathcal{N} , une formule CTL φ sur le graphe de régulation GR associé, et une paramétrisation $P \in \mathcal{PN}$, nous dirons que P est un modèle de φ ssi $check_{\mathcal{N}}(P, \varphi) = true$. L'ensemble des modèles de φ est noté $\mathcal{M}(\varphi) = \{P \in P_{\mathcal{N}} \mid check_{\mathcal{N}}(P, \varphi) = true\}$.

3.3.2.1 Les limites de CTL

Supposons que nous souhaitions vérifier qu'une variable x va toujours aller de manière asymptotique à son niveau le plus haut, soit le niveau 2 dans cet exemple. Pour représenter cette information, on souhaiterait naturellement écrire la formule CTL $AF(AG(x = 2))$. Considérons le graphe de transitions de la figure 3.12.

A l'état (1,0) il y a deux possibilités :

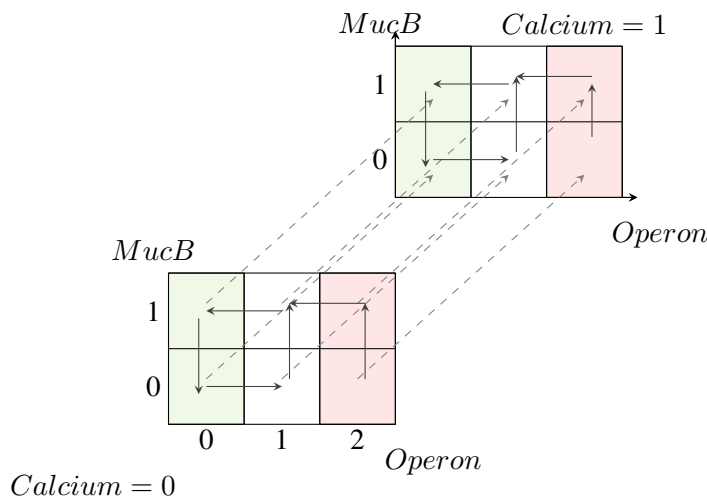


Figure 3.11 – Graphe de transitions des états pour *Pseudomonas aeruginosa*. Les cases rouges ($Operon=2$) indiquent les états à ne jamais atteindre pour satisfaire la formule, tout en partant des cases vertes ($Operon=0$).

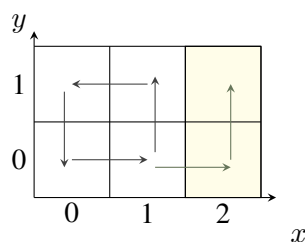


Figure 3.12 – Graphe de transitions montrant la non équitabilité des chemins en CTL . Le chemin infini qui boucle sur les états blancs n’atteindra jamais un état où $x=2$.

- soit x augmente et le système va atteindre l’état $(2,0)$ et y rester,
- soit y augmente et le système peut rester indéfiniment dans les états blancs suivants : $(0,0)$, $(0,1)$, $(1,0)$ et $(1,1)$.

La formule $AF(AG(x = 2))$ est donc évaluée à faux dans chacun des états blancs puisqu’il existe un chemin qui n’atteindra jamais un état où $x = 2$. Le seul chemin qui contredit la formule est celui où à chaque fois que l’on passe dans l’état $(1,0)$ on choisit la transition qui fait augmenter y ; la transition qui fait augmenter x n’est jamais choisie. En ce sens, la sémantique de CTL n’est pas équitable puisqu’on passe une infinité de fois dans l’état $(1,0)$ sans jamais choisir la transition qui fait augmenter x .

D’un point de vue biologique il est possible d’observer de *manière transitoire* une oscillation avant d’atteindre l’état d’équilibre (ou un bassin d’attraction). C’est finalement, ce type de comportements asymptotiques que CTL ne capture pas de « manière naturelle ». CTL introduit donc un artefact non souhaité par rapport aux comportements biologiques réels. Une solution efficace consiste à modifier la sémantique de CTL en ignorant les chemins non équitables. Pour y parvenir, nous utiliserons la sémantique *Fair-Path CTL*, définie par Adrien Richard [Richard, 2008].

3.3.2.2 *Fair-Path CTL*

En *Fair-Path CTL*, les quantificateurs de chemins A et E sont respectivement convertis en A' et E' . La sémantique de ces nouveaux quantificateurs est équitable par définition : le système ne peut pas entrer une infinité de fois dans un état sans emprunter au moins une fois toutes les transitions possibles sortantes pour cet état. De cette manière il est possible de représenter l'ensemble des comportements asymptotiques équitables des systèmes biologiques. De plus il a été montré que cette sémantique est traduisible en *CTL* [Richard, 2008], autrement dit que l'on peut écrire une formule *CTL* sémantiquement équivalente (mais notablement plus longue).

3.3.3 Matrices de phénotypes

Lorsque l'on représente des systèmes biologiques de grande envergure beaucoup de connaissances biologiques sont disponibles pour décrire le phénotype du système. Chaque phénotype peut être normalisé en *Fair-Path CTL*. Dans ces conditions le nombre de formules associées sera très grand : il faut donc un moyen de les gérer comme des éléments majeurs de spécification.

On remarque par ailleurs que toutes les variables n'ont pas le même rôle dans un graphe de régulation. Certaines variables comme *Calcium* dans le graphe de régulation de la production de mucus chez *Pseudomonas aeruginosa* (Fig. 3.1) n'ont pas de prédécesseurs : ce sont des variables sources (qui ont un degré entrant 0). Une variable source n'a qu'un unique paramètre associé : elle tend donc vers la valeur de son unique paramètre. Une variable source pourra être considérée comme constante, si l'on regarde son comportement asymptotique. Ce sont en fait, des variables qui traduisent l'environnement du système biologique. En effet, en jouant avec les valeurs de ces paramètres spéciaux (paramètres des différentes variables sources) on pourra constituer différentes configurations du système biologique. Chacune de ces configurations indique une combinaison particulière de ressources dans lequel le système peut se retrouver, que l'on appellera *un environnement*. Pour chaque environnement, le comportement du système est différent, classifier les comportements en fonction de ces environnements est donc souhaitable.

Enfin, de manière empirique la majorité des connaissances sur les comportements ne mettent en jeu qu'un nombre très restreint de variables. On sait par exemple que telle ou telle variable va osciller, tendre vers une valeur, *etc.* On peut donc aussi classer les comportements *via* la variable principale à laquelle ils se rapportent.

Au final, l'ensemble de ces informations conduit de manière naturelle à regrouper les connaissances biologiques formalisées dans une matrice dont les lignes sont les différents états possibles des variables sources et où les colonnes sont dédiées à chacune des autres variables. Chaque case indiquant le comportement d'une variable dans un environnement précis. Ce type de matrice est une **matrice de phénotypes**.

Dans ce chapitre nous avons vu l'ensemble des éléments nécessaires pour construire un réseau de régulation, comment est décrite la dynamique de ce réseau et comment vérifier sa cohérence par rapport aux propriétés biologiques du système à modéliser. L'ensemble des travaux présentés dans ce manuscrit repose sur ce cadre formel qui permet de confronter des informations structurelles sur les interactions (graphe de régulation) aux informations comportementales connues sur le système étudié. La partie suivante présente plus en détail une méthodologie de développement de modèles

à grande envergure en deux étapes qui s'appuie sur la vérification formelle de propriétés provenant d'une matrice de phénotypes.

Cycles de vie pour les modèles formels de systèmes dynamiques

CHAPITRE 4

Cycles de vie : du génie logiciel aux modèles de systèmes dynamiques

Ce chapitre introduit deux cycles de vie pour les modèles formels de systèmes dynamiques. Le premier est inspiré du développement en « V » classiques dans les domaines technologiques. Il décrit les étapes de développement d'un premier modèle en partant seulement des connaissances sur le système biologique à représenter. Le second est inspiré des cycles de vies en spirale qui réutilisent de précédents modèles en les enrichissant cycle après cycle.

4.1	Méthodologie de création de modèles <i>de novo</i>	41
4.1.1	Définition d'une question biologique	42
4.1.2	Construction du contexte de modélisation	42
4.1.3	Obtention des graphes de transitions crédibles	43
4.1.4	Obtention des graphes de transitions corrects	43
4.2	Méthodologie de modélisation par enrichissements successifs . .	44
4.2.1	Démarche générale	44
4.2.2	Phases d'un enrichissement	44

Ce chapitre a pour but de poser notre contexte méthodologique de développement de modèles de systèmes dynamiques. Nous décrivons une méthode rigoureuse de développement de modèles *ex nihilo*, puis la méthode développée au cours de cette thèse pour l'obtention de modèles conséquents [Gibart et al., 2021a]. Ces deux méthodes ne sont pas sans rappeler des techniques classiques du génie logiciel et nous nous efforcerons d'en expliciter les similitudes et les différences.

4.1 Méthodologie de création de modèles *de novo*

La première méthode de développement de modèle consiste à développer un modèle en partant de zéro, sans aucune autre base que des connaissances biologiques, et en spécifiant les besoins généraux. Comme le cycle de vie en « V » [3] bien connu en technologie, le développement d'un modèle formel de système dynamique se fait en deux phases principales (voir schéma 4.1) :

- Une phase descendante qui détaille le produit *via* ses spécifications, jusqu'à sa réalisation, qui correspondra à la création de la dynamique dans notre cas
- Puis une phase ascendante qui a pour but la vérification *via* des méthodes formelles jusqu'à validation par les modélisateurs et biologistes¹.

4.1 cycle de vie en "V"

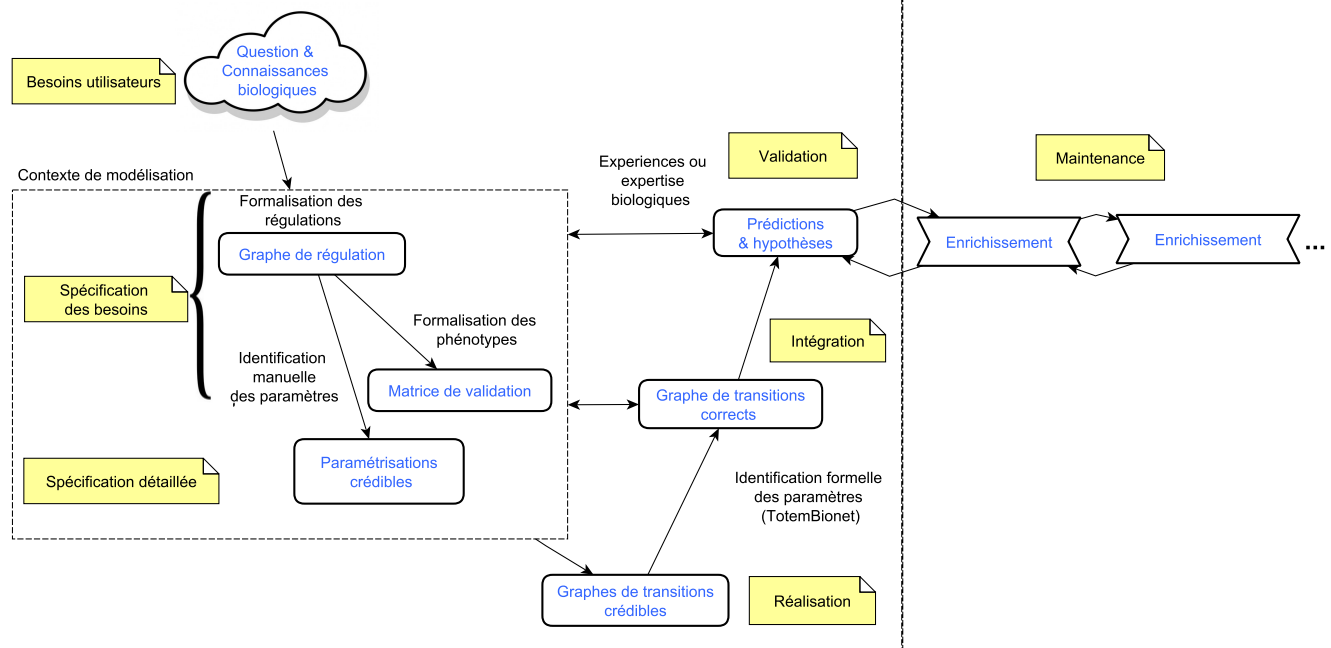


Figure 4.1 – Cycle de vie en « V » adapté au développement de modèles de systèmes dynamiques. En bleu les objets formels construits, en noir les activités et en jaune les analogies avec le génie logiciel.

1. Dans notre cadre, faire une distinction précise entre *vérification* et *validation* n'aurait en fait pas grand sens. En Génie logiciel vérifier consiste à s'assurer que les spécifications sont assurées par l'outil développé, tandis que la validation consiste à s'assurer que les résultats produits par l'outil sont bien atteints

Dans cette partie, nous décrivons étape par étape le développement d'un modèle *de novo* en s'appuyant sur la partie gauche de la figure 4.1.

4.1.1 Définition d'une question biologique

Le développement d'un modèle démarre lorsqu'un chercheur en biologie se pose une question sur les régulations d'un système biologique à laquelle il n'est pas capable de répondre directement, mais où la modélisation peut l'aiguiller dans la bonne direction. A partir de cette question et des connaissances des biologistes, qui correspondraient aux besoins des utilisateurs d'un point de vue génie logiciel, nous sommes capables de définir un contexte de modélisation.

4.1.2 Construction du contexte de modélisation

La construction du contexte de modélisation passe par la conception des trois éléments (figure 4.2) : le graphe de régulation (en jaune), la matrice de validation (blanc) et la mise en place de paramétrisations crédibles biologiquement (en vert).

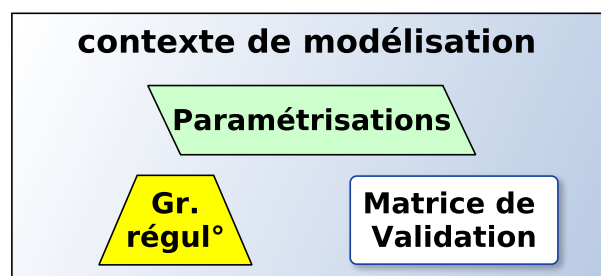


Figure 4.2 – Schéma d'un contexte de modélisation

La formalisation du *graphe de régulation* est la première étape de la construction du contexte de modélisation, c'est une étape de spécification des connaissances biologiques qui s'apparente à l'analyse des besoins en génie logiciel. A partir des connaissances statiques sur le système à étudier et de la question biologique, on définit les entités et les régulations qui les lient comme on l'a vu dans le chapitre 3.

Ensuite nous formalisons les connaissances que l'on a sur la dynamique du système. Ces informations sont rangées en deux catégories :

les connaissances globales : portent sur les phénotypes du système pris dans son ensemble.

Elles permettent de constituer une matrice de phénotype où tous les comportements indiqués font consensus. Cette matrice est appelée *matrice de validation* (décrite chapitre 5),

les connaissances détaillées : correspondent aux informations dynamiques que l'on a sur les interactions locales entre les différentes entités du système. Dans la théorie de Thomas elles permettent d'identifier manuellement les paramètres *via* les expériences de pensée (décrites chapitre 6) et d'obtenir les *paramétrisations crédibles*.

La formalisation des phénotypes et l'identification des paramètres peuvent être menées en parallèle. La première activité, qui mène à une matrice de validation, n'est pas sans rappeler les matrices de tests fonctionnels extraites des spécifications en génie logiciel. La deuxième activité qui consiste à identifier partiellement les paramètres peut être vue comme une activité de raffinement des spécifications (elle raffine le graphe de régulation par des indications sur la force relative des interactions).

4.1.3 Obtention des graphes de transitions crédibles

En partant de paramétrisations crédibles du contexte de modélisation, nous définissons ensuite des graphes de transitions crédibles. Ces graphes correspondent à une représentation des événements dynamiques locaux crédibles mis bout à bout pour obtenir des dynamiques globales crédibles pour le contexte proposé. En ce point, si nous continuons le parallèle avec le génie logiciel, nous avons terminé toutes les spécifications et nous arrivons au plus bas du cycle de vie en « V » : l'étape de réalisation.

4.1.4 Obtention des graphes de transitions corrects

A cette étape de réalisation nous avons à disposition une représentation des dynamiques générales crédibles *via* les graphes de transitions déduits des paramétrisations crédibles (cf. chapitre 3, section 3.2). Nous disposons de plus de la spécification des comportements globaux attendus *via* la matrice de validation. Parmi tous les graphes de transitions crédibles, issus des paramétrisations crédibles, certains vont exhiber des comportements globaux en contradiction avec la matrice. Ceux qui exhibent les comportements attendus sont les *graphes de transitions corrects*. L'enjeu de cette première étape dans la remontée du cycle en « V » est d'obtenir formellement ces graphes de transitions corrects. Ils sont cohérents avec les spécifications phénotypiques globales données dans la matrice de validation. L'identification formelle de ces paramétrisations se fait à l'aide de *TotemBioNet* (outil que l'on présente dans le chapitre 6, qui traite de l'identification des paramètres).

Obtenir les graphes de transitions corrects constitue donc d'une certaine façon l'activité d'intégration d'un point de vue génie logiciel. Si pendant cette phase aucun graphe de transition ne s'avère correct, cela nous indique que l'un des composants du contexte de modélisation doit être réévalué (d'où la présence de la flèche liant graphe de transitions correct et contexte de modélisation dans la figure 4.2).

4.1.4.1 Prédications et hypothèses

Dans la majeure partie des cas, plusieurs graphes de transitions (*i.e.* plusieurs paramétrisations) sont corrects pour un contexte de modélisation donné. On considérera que chaque dynamique correcte est un modèle². Chaque modèle possède sa propre dynamique et les différences entre les dynamiques des modèles constituent des prédictions pouvant mettre en avant des hypothèses à propos de la question biologique initiale. L'analyse de ces prédictions et hypothèses se fait avec l'aide d'un biologiste qui, à l'aide de son expertise et/ou d'expériences, valide le contexte de modélisation créé. Cette étape constitue l'activité de validation.

2. adoptant en cela la terminologie de la logique mathématique.

Une fois validé, le contexte de modélisation établi a rempli sa fonction, il n’y a donc pas à proprement parler d’activité de maintenance comme l’on pourrait le voir pour le développement d’un logiciel. Cependant, pour répondre à d’autres problématiques liées au même système biologique ou lorsque de nouvelles connaissances apparaissent, il est possible de réévaluer le contexte de modélisation. Dans ce cadre, plusieurs enrichissements successifs peuvent lui être apportés. Malheureusement, les étapes d’enrichissement d’un contexte de modélisation s’avèrent beaucoup plus complexes qu’en génie logiciel car les modèles de réseaux biologiques sont intrinsèquement non modulaires contrairement aux logiciels. Pour pallier ce problème, nous développons dans cette thèse une nouvelle méthode d’enrichissement incrémentale qui permet d’assurer la non-régression des spécifications initiales, tout en proposant des améliorations au contexte de modélisation initial. C’est le sujet de la section suivante.

4.2 Méthodologie de modélisation par enrichissements successifs

Une fois le contexte de modélisation validé en suivant la méthodologie de développement de modèles *ex nihilo* inspirée du cycle de vie en « V », nous proposons une méthodologie d’enrichissement incrémentale inspirée des méthodes de développement en spirale [Boehm, 1988]. Les enrichissements successifs que propose notre méthodologie permettent notamment d’adapter un contexte de modélisation à une nouvelle question biologique, pourvu que le système biologique d’intérêt soit déjà formalisé en grande partie par ce contexte de modélisation.

4.2.1 Démarche générale

Une des problématiques majeures pour développer des modèles de cette manière est la conservation de propriétés dynamiques. En effet, l’ajout ou la modification de composants dans le graphe de régulations, la modification de la valeur d’un paramètre peut avoir un impact considérable comme dans tout système complexe. Pour s’affranchir de ce problème d’absence d’incrémentalité, le développement en spirale est une approche rationnelle. Le principe est de répéter les opérations d’analyse des besoins, d’enrichissement et vérification du contexte de modélisation nouvellement créé puis d’avoir un retour prédictif à chaque étape d’enrichissement (cf. figure 4.3). De cette façon, l’ampleur de chaque modification incrémentale peut être limitée presque à volonté, de sorte que les preuves de non régression majeures dans ce contexte, restent faisables en un temps raisonnable.

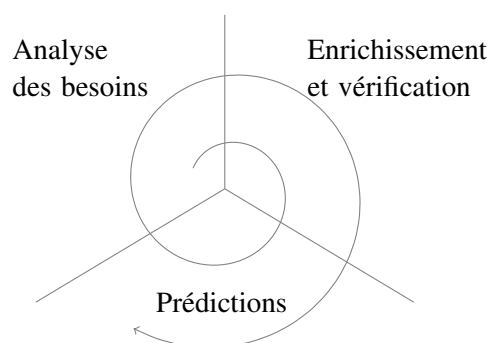


Figure 4.3 – Cycle de vie en spirale pour le développement de modèles enrichis

4.2.2 Phases d’un enrichissement

Comme l’utilisation et la mise en place de cette méthodologie est un apport informatique méthodologique important de notre travail, il convient de décrire son processus général, puis de

détailler cette méthodologie *via* différents chapitres dans le manuscrit. La figure 4.4 décrit schématiquement les trois phases que l'on suit pour enrichir un modèle. Ces phases sont les mêmes que celles indiquées dans le cycle en spirale et se font dans l'ordre suivant : (1) Analyse des besoins ; (2) enrichissement et vérification ; (3) prédictions.

4.2.2.1 Analyse des besoins

Dans un premier temps à partir d'un contexte de modélisation initial, nous définissons les différents besoins liés à la nouvelle question biologique traitée (en haut de la figure 4.4). Cette analyse implique une collaboration étroite entre biologistes et modélisateurs. Le rôle du biologiste est d'indiquer tous les éléments biologiques qui ne sont pas représentés mais qui sont nécessaires pour répondre à sa problématique, et le modélisateur avec sa connaissance du modèle, va cibler les différents enrichissements qu'il devra faire et les nouvelles entités formelles que cela implique (variables, multiplexes, ...).

4.2.2.2 Enrichissement et vérification

La phase d'enrichissement est alors divisée en deux parties : la première consiste à formaliser les éléments dans les différents composants du contexte de modélisation (graphe de régulation, paramètres, matrice de validation) et la seconde à vérifier la cohérence de la formalisation proposée de manière formelle (au centre de la figure 4.4).

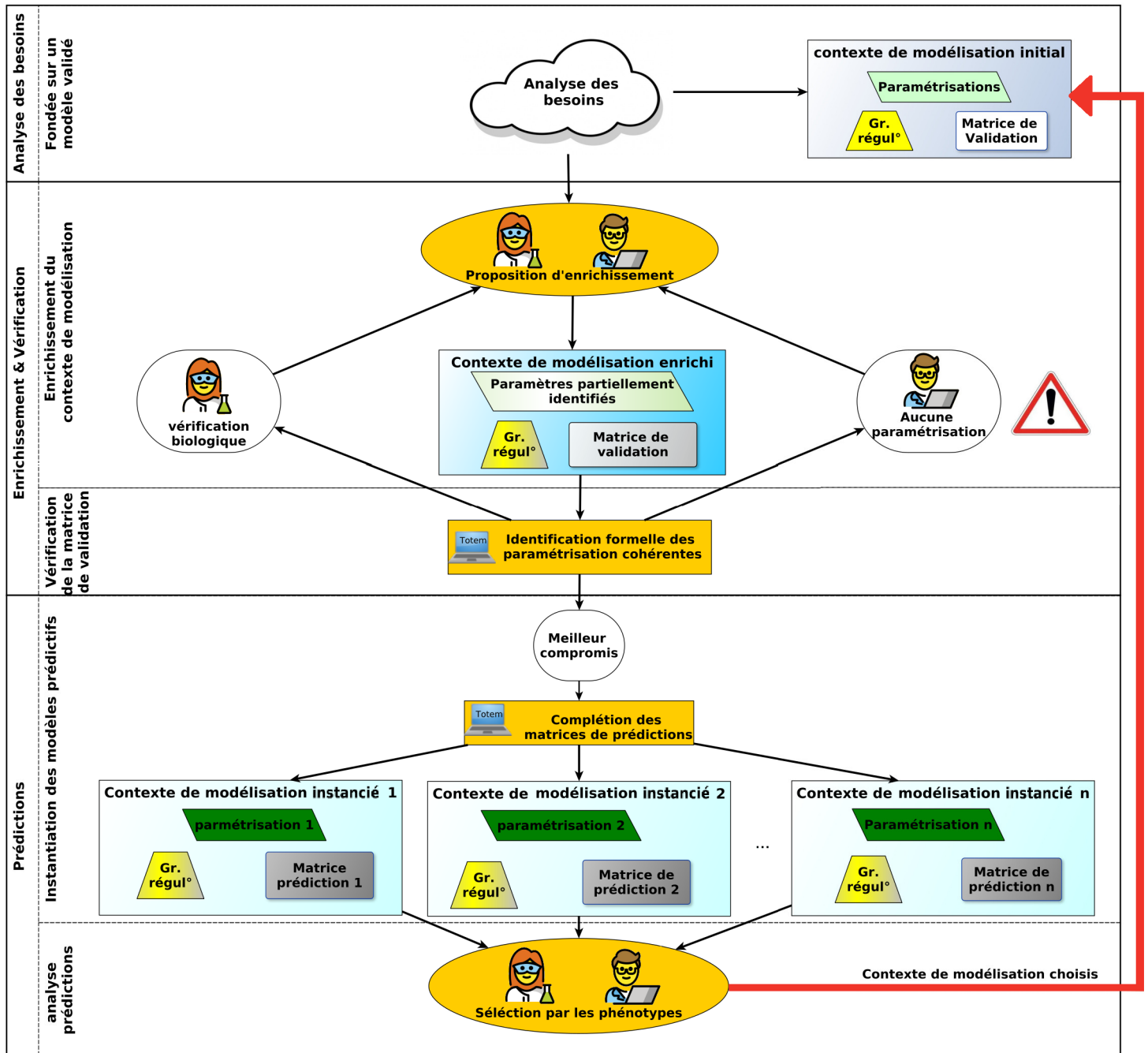
Enrichissement. A partir de la description faite par les biologistes et du contexte de modélisation initial, le modélisateur va formaliser :

- les nouvelles connaissances statiques *via* la modification du graphe de régulation,
- les nouvelles connaissances sur les dynamiques locales du système encodé *via* l'identification manuelle des paramètres
- et les nouvelles connaissances sur les phénotypes globaux formalisées *via* la réévaluation de la matrice de validation

Différents types d'enrichissement peuvent être effectués sur un contexte de modélisation. Leur description et leur mise en place sont détaillées plus loin dans le chapitre 8 de ce manuscrit.

Validation. Une fois l'enrichissement du contexte de modélisation terminé, nous devons vérifier qu'il existe des paramétrisations qui induisent des graphes de transitions corrects vis à vis de la matrice de validation. Comme pour le développement de modèle *ex nihilo*, après vérification on peut distinguer deux cas de figure :

- Si aucune paramétrisation ne fournit de graphe de transition correct, alors, on doit reconsidérer tout ou partie du contexte de modélisation enrichi. On doit donc réexaminer les connaissances biologiques pour redéfinir un contexte de modélisation enrichi différent. Cette révision peut conduire à remettre en cause une « connaissance » biologique et suggérer de nouvelles expériences en laboratoire.
- S'il existe des paramétrisations qui induisent des graphes de transition corrects, on entre dans la phase de prédiction et retour à la biologie.



Méthodologie incrémentale d'enrichissement de modèles

Figure 4.4 – Méthodologie incrémentale d'enrichissement de modèles.

4.2.2.3 Prédiction

La phase de prédiction est elle aussi en deux phases : l’instanciation des modèles et l’analyse des prédictions (en bas de la figure 4.4).

L’instanciation des modèles. La première phase consiste à instancier les différents modèles obtenus par l’étape précédente. Instancier les modèles consiste à créer une matrice de prédiction pour chaque paramétrisation correcte avec le contexte de modélisation enrichi. Une matrice de prédiction est l’inventaire exhaustif des phénotypes globaux prouvés formellement *via TotemBioNet*. Des comportements encore inconnus sont donc consignés dans ces matrices.

Analyse des prédictions. Une analyse des différentes matrices de prédictions permettra aux biologistes et au modélisateur d’éliminer les comportements aberrants et d’extraire à l’inverse des propriétés intéressantes suite à l’enrichissement. Souvent le nombre de modèles qui restent acceptables est grand, ce qui peut conduire à choisir les paramétrisations les plus pertinentes. Ces focalisations permettent de réduire les domaines des paramètres dans le contexte de modélisation.

Une fois ces trois étapes terminées, l’enrichissement (un tour du cycle en spirale) est terminé et nous obtenons un contexte de modélisation cohérent. Ce dernier pourra être réutilisé pour une nouvelle phase d’enrichissement (modélisé dans le schéma par la flèche rouge).

Ces trois étapes sont très importantes pour prouver la non-régression. En effet, l’ajout d’un composant au réseau de régulation, aussi mineur ou périphérique soit-il, peut drastiquement modifier la dynamique du modèle. Passer par de petits enrichissements successifs tout en conservant les spécifications est un moyen de faire face à ces problèmes de mise au point en réduisant chaque étude à un nombre très limité de nouvelles interactions dans le graphe. Cela facilite les preuves de non-régression.

Au cours de ce chapitre nous avons vu deux approches complémentaires permettant d’élaborer des modèles de grande envergure, en construisant un premier modèle *ex nihilo* puis en enrichissant ce modèle cycle après cycle. La première méthode a été utilisée pour décrire un modèle général de la régulation du métabolisme décrite au chapitre 9, puis nous proposons deux adaptations en plusieurs cycles d’enrichissement dans les chapitres 10 et 11. Ces deux chapitres décrivent les modèles de la régulation du métabolisme que nous avons construits dans le cadre l’adénocarcinome ductal du pancréas (PDAC), puis plus spécifiquement celui du PDAC agressif.

Les chapitres suivants décrivent pas à pas et en détail les différentes étapes, pour la mise en place et l’enrichissement d’un modèle. Le chapitre 5 décrit la collecte et la formalisation des connaissances biologiques pour la construction du graphe de régulation et de la matrice de validation. Le chapitre 6 met en avant les deux méthodes (manuelles et formelles) d’identification des paramètres dynamiques. Le chapitre 7 propose une méthode pour vérifier formellement la cohérence du contexte de modélisation créé ou enrichi *via TotemBioNet*. Le chapitre 8 détaille les étapes à suivre pour formaliser l’enrichissement d’un contexte de modélisation.

CHAPITRE 5

Analyse du contexte, des principaux acteurs et des comportements attendus

Nous décrivons dans ce chapitre les premières étapes permettant de créer un contexte de modélisation, en passant par une phase d'analyse des besoins. Le modélisateur formalisera toujours en premier les régulations, puis il formalisera les phénotypes. Ces deux étapes constituent un contexte de modélisation qu'on peut assimiler aux spécifications des besoins en génie logiciel.

5.1	Formalisation des régulations	49
5.1.1	Les variables	49
5.1.2	Les Multiplexes	50
5.2	Formalisation des phénotypes	50
5.2.1	Marqueurs et environnement	50
5.2.2	Propriétés environnementales	51
5.3	Un langage pour les phénotypes courants	52
5.3.1	Comportements stationnaires	52
5.3.2	Comportements oscillatoires	53
5.3.3	Des motifs à la demande	54
5.4	Formalisation de la matrice de phénotypes	55
5.4.1	Structure matricielle	55
5.4.2	Illustration avec <i>Pseudomonas aeruginosa</i>	56

Comme nous l'avons abordé dans le chapitre précédent, le développement d'un modèle peut s'apparenter aux activités classiques en génie logiciel. Dans ce chapitre on spécifie ce qu'on attend d'un modèle dynamique formel, en d'autres termes, on discute de la formalisation des régulations *via* le graphe de régulation et de la formalisation des phénotypes attendus *via* la création d'une matrice de phénotypes.

5.1 Formalisation des régulations

Pour représenter différentes régulations, on commence par se donner une représentation statique commune pour formaliser les éléments qui vont paraître dans le graphe de régulation. En génie logiciel, on apparenterait cette étape à la mise en place d'une signature pour une spécification formelle. Formellement, un graphe de régulation est constitué de variables et de multiplexes. Définir quels concepts biologiques peuvent représenter ces deux éléments est important pour avoir une formalisation cohérente des régulations.

5.1.1 Les variables

D'un point de vue biologique on peut considérer deux types de variables :

- Les Variables Internes, appartenant à un ensemble VI , qui correspondent à des composés ou groupes de composés biologiques, régulant et régulés par le système et pour lesquels on souhaite étudier ou observer le comportement.
- Les Variables Environnementales, appartenant à un ensemble VE , qui représentent l'environnement du système. Ces variables ne sont régulées par aucun élément interne au système. Cependant, leurs états jouent un rôle sur le comportement global du système car elles en régulent certaines variables internes. Ce sont les variables d'entrée du système, elles peuvent représenter des concepts comme la température, la présence d'un nutriment ou un élément biologique transmettant un signal à la cellule. Ces variables s'apparentent aux conditions expérimentales que l'on peut retrouver en laboratoire de biologie. Elles seront donc fixées pour les expériences et permettront de placer le système dans une configuration précise.

Un graphe de régulation sera donc légèrement enrichi par rapport à la notion donnée dans le chapitre 3, pour prendre en compte cette nuance.

Définition 5.1.1 (Graphe de régulation avec variables d'environnement). *Un graphe de régulation avec variables d'environnement $G = (V, VE, M, A)$ est un graphe de régulation tel que :*

- (V, M, A) est un graphe de régulation simple,
- $VE \subsetneq V$ est un ensemble ordonné de variables d'environnement,
- Les variables d'environnement n'ont pas de prédécesseurs dans $G : \forall v \in VE, d^-(v) = 0$,
- Les variables d'environnement peuvent apparaître dans les atomes des formules de multiplexes.

Nous notons VI l'ensemble des variables internes : $VI = V \setminus VE$.

Chaque variable d'environnement v , n'ayant pas de prédécesseurs, possède un unique paramètre K_v dont la valeur sera par la suite fixée égale à son état initial. Un environnement est donc

défini par la famille $\{K_v\}_{v \in VE}$ et le mécanisme assurant la stabilité des variables d'environnement est implémenté dans l'outil *TotemBioNet* permettant l'énumération des paramétrisations et la vérification formelle des contextes de modélisation décrits dans le prochain chapitre.

Exemple 5.1.1 – En reprenant le graphe de régulation de *Pseudomonas aeruginosa* présenté chapitre 3, figure 8.1, la variable en vert, **calcium** est une variable d'environnement. Le paramètre $K_{Calcium}$ associé aura la valeur de l'état initial de **Calcium**. Les autres variables, **operon** et **MucB** elles sont des variables internes.

5.1.2 Les Multiplexes

Les multiplexes vont abstraire différents concepts biologiques. Dans le sens commun un multiplexe sert à représenter une coopération entre différentes entités biologiques pour encoder une régulation. Biologiquement ces régulateurs correspondent à des molécules ou ensembles de molécules biologiques. Conceptuellement, une variable et un multiplexe ne sont différenciées d'un point de vue biologique, que par leur « rôle » dans le graphe. Un élément abstrait par une variable interne pourra donc être aussi bien représentée par un multiplexe. Ce choix relève de la conception du graphe de régulation. Néanmoins, lors de la formalisation d'un graphe de régulation, il faut être conscient que la représentation d'un élément par un multiplexe n'est judicieuse que si cet élément est de nature booléenne et que son statut de ressource ou non ressource de ses cibles est « immédiat ». En effet, un multiplexe n'a pas d'états, donc sa « présence » est conséquence immédiate de l'état des variables auxquelles il fait appel. Ce qui n'est pas le cas lorsque l'on représente un élément biologique par une variable, car une variable possède des états qui peuvent être mis à jour uniquement après plusieurs étapes en raison de la concurrence entre les variables qui peuvent être mises à jours. Par conséquent, comme les entités de régulation représentées par un multiplexe ne possèdent pas d'états, leur impact d'un point de vue comportemental est légèrement différent que si on les avait représenté ces entités par des variables.

Pragmatiquement, les variables sont par défaut les concepts majeurs du système biologique représenté, les multiplexes encodent les conditions connues d'influence des variables. On examine ensuite chaque variable booléenne pour déterminer si son état doit être explicite ou s'il peut devenir une « simple » valeur de vérité, auquel cas cette variable peut potentiellement devenir un multiplexe.

5.2 Formalisation des phénotypes

La formalisation des phénotypes consiste en la traduction formelle des comportements connus des marqueurs biologiques en fonction de l'environnement.

5.2.1 Marqueurs et environnement

Biologiquement un *marqueur* est une molécule ou ensemble de molécules pour lequel on a, au préalable, expérimentalement établi l'évolution au cours du temps. Par abus de langage, lorsqu'on traduira le comportement d'un marqueur biologique, on utilisera son équivalent *formel* qui est représenté, le plus souvent, dans le graphe de régulation par une variable interne.

Exemple 5.2.1 – Dans le graphe de régulation de *Pseudomonas aeruginosa*, **Operon** représente à la fois un élément génétique activable (l’opéron) et la protéine exprimée qui forme le mucus lorsque l’opéron est activé à son niveau maximal (2). Une bactérie peut avoir un phénotype mucoïde ou non. Lorsqu’elle est mucoïde, elle produira du mucus. La production de mucus peut être suivie expérimentalement. Le mucus est donc un marqueur biologique. La description formelle de l’état mucoïde ou non mucoïde nécessitera l’utilisation de la variable **Operon** : la variable *operon* est l’équivalent *formel* du marqueur biologique.

Le comportement de ces marqueurs dépend évidemment de l’environnement. L’environnement est représenté par une combinaison particulière de valeurs des variables d’environnement : une température spécifique, une disponibilité particulière en nutriments, en lumière *etc.* Réussir à formaliser correctement les phénotypes nécessite donc de définir formellement ce qu’est un environnement à partir des variables d’environnement.

Environnements Formellement les environnements ont représentés par l’intermédiaire des variables d’environnement présentes dans le graphe de régulation. Lorsqu’on instancie les variables d’environnement, on se place dans une configuration particulière pour le réseau qui correspond à un environnement. Un environnement est donc représenté par une suite de valeurs associées une à une aux différentes variables d’environnement ordonnées. Plus on aura de variables d’environnement avec des domaines d’applications étendus, plus le nombre de configuration du réseau sera grand et par extension le nombre d’environnements aussi.

Définition 5.2.1 (Environnements). *Étant donné un graphe de régulation avec environnements* $G = (V, VE, M, A)$, un environnement e est une substitution $e : VE \rightarrow \mathbb{N}$, telle que $\forall v \in VE, e(v) \leq b_v$ où b_v est la borne maximum de v . Alternativement, on peut considérer que e est un n -uplet avec $n = |VE|$. Enfin, on note E l’ensemble des environnements. Le nombre total d’environnement est $|E| = \prod_{v \in VE} (b_v + 1)$.

Exemple 5.2.2 – Pour *Pseudomonas aeruginosa* nous avons une variable d’environnement (**Calcium**) qui peut prendre la valeur 0 ou 1. L’ensemble E des environnements est donc $E = \{e^0, e^1\}$. Nous serons dans l’environnement e^0 lorsque *Calcium* = 0 et e^1 lorsque *Calcium* = 1.

5.2.2 Propriétés environnementales

Maintenant que nous avons fait correspondre les marqueurs à des variables internes et représenté l’environnement au travers des variables d’environnements, nous avons toutes les clefs pour traduire les comportements à l’aide de formules logiques (un exemple est donné section 3.3). La formalisation du comportement d’une variable interne pour chaque valuation des variables d’environnement constitue une propriété environnementale.

Définition 5.2.2 (Propriété environnementale). Une propriété environnementale est un couple (e, φ) où φ est une formule *Fair-Path* CTL dans \mathcal{L}_{VI} , le langage des formules *Fair-Path* CTL sur les variables internes dans VI et $e \in E$ est un environnement. φ exprime une propriété temporelle biologique qui est observée dans les conditions environnementales représentées par e .

Nous choisissons de définir les propriétés environnementales *via* une formule *Fair-Path CTL* car la majorité des comportements des marqueurs s’observe lorsque le système à atteint un régime stable et qu’avec cette sémantique on s’affranchit des problèmes éventuels liés aux comportements transitoires (voir explications section 3.3.2.2, page 37).

Exemple 5.2.3 – Comme indiqué dans l’exemple précédent, le réseau de *Pseudomonas aeruginosa* possède deux environnements e^0 et e^1 .

- Dans e_0 (sans calcium), la bactérie exhibe deux phénotypes stables : une bactérie mucoïde (resp. non mucoïde) restera mucoïde (resp. non mucoïde). Sachant que le niveau 0 pour de **Operon** correspond à un état non mucoïde et que le niveau 2 à un niveau Mucoïde, pour illustrer l’affirmation précédente on utilise la formule¹ :

$$\varphi^0 \equiv \begin{aligned} &(((Operon = 2) \Rightarrow A'G((Operon = 2))) \\ &((Operon < 2) \Rightarrow A'G((Operon < 2)))) \end{aligned}$$

- Dans e^1 *Pseudomonas aeruginosa* devient toujours mucoïde, ce qui est exprimé par la formule :

$$\varphi^1 \equiv A'G(A'F(Operon = 2))$$

Ainsi, il existe deux propriétés environnementales (e^0, φ^0) et (e^1, φ^1) .

Au final l’ensemble des comportements des marqueurs d’un système sera représenté par une liste de propriétés environnementales : $\Psi = [(e^i, \varphi^i)]_{i=1}^n$ où $n = |E|$

5.3 Un langage pour les phénotypes courants

Lorsque l’on s’intéresse aux comportements quand le régime est stabilisé on retrouve deux grandes classes de comportements communs : les comportements stationnaires et les comportements oscillatoires. Comme on doit formaliser des comportements asymptotiquement stables du système, on devra écrire des formules temporelles qui s’affranchissent de la partie transitoire des traces temporelles.

5.3.1 Comportements stationnaires

Biologiquement, on considère que l’on a un comportement stationnaire lorsque la concentration du marqueur devient stable et le reste suffisamment longtemps (notion d’état stable). Dans le cadre du formalisme de René Thomas, ce comportement stationnaire est un état du graphe de transitions asynchrone sans transition sortante.

Exemple 5.3.1 – Prenons l’exemple du graphe de transitions donné figure en 3.12. On constate que tous les chemins mènent à l’état où $x = 2$ et $y = 0$. Aucun arc ne sort de cet état : c’est donc un état stationnaire. Par conséquent la variable x atteint 2 au bout d’un certain nombre de transitions et y reste.

1. pour rappel A' et E' sont des notations *Fair-Path CTL*.

Fair-Path CTL pour les comportements stationnaires. On ne considère le système qu'une fois qu'il a atteint un régime stationnaire. On ignore donc les états transitoires du système et ce, quelle que soit la trace suivie. Or, nous avons vu dans le chapitre 3 que le langage *CTL* avait quelques limites, car s'il existe une trace qui fait osciller transitoirement une variable avant d'atteindre un équilibre au régime stationnaire, *CTL* n'est pas capable de faire abstraction de cette phase transitoire. Comme nous l'avons aussi décrit dans le chapitre 3, *Fair-Path CTL* est une solution car c'est une variante qui permet l'abstraction de tous les états transitoires, oscillations comprises. On utilise donc cette sémantique pour décrire les comportements stationnaires de manière générale.

Exemple 5.3.2 – Biologiquement on sait que lorsqu'il y a du calcium, la bactérie deviendra toujours mucoïde. Pour rappel, l'état mucoïde de la bactérie est représenté par la présence de mucus, elle même associée à l'activité de l'opéron. On décrit donc l'état mucoïde de la bactérie via le niveau de la variable *operon*.

Pour spécifier le comportement de l'Opéron en présence de calcium, il faut donc dire que le système atteint un comportement stationnaire de la forme *Operon* = 2 et y rester. Autrement dit, on souhaite atteindre dans le futur pour tous les chemins possibles un état où *Operon*=2, et à partir de là on souhaite que cette propriété là reste vraie. Cette propriété s'écrit :

$$A'F(A'G(\text{Operon} = 2))$$

En généralisant l'exemple on peut donner une définition d'un comportement stationnaire :

Définition 5.3.1 (Comportement stationnaire). *Étant donné une variable interne $v \in VI$ et un entier naturel $n \in \llbracket 0, b_v \rrbracket$, on note $td_v(n)$ le comportement stationnaire de v à la valeur n . La formule *Fair-Path CTL* suivante :*

$$td_v(n) \equiv A'F(A'G(v = n))$$

Lorsque cette formule est satisfaite on dit que v tend vers n .

5.3.2 Comportements oscillatoires

Dans un comportement oscillatoire, le marqueur ne va pas se fixer, il aura toujours une trajectoire oscillante entre plusieurs états, et ce même une fois que le système est globalement arrivé à un phénotype stable. Dans le cadre du formalisme de René Thomas ce comportement oscillatoire donne bien lieu, dans le graphe de transitions asynchrone, à au moins un cycle.

Exemple 5.3.3 – Reprenons le graphe de transitions proposé dans la figure 3.5 provenant de l'exemple 3.2.5. En partant de n'importe quel état, on constate que l'on peut arriver dans le cycle constitué des 4 états suivants : $(0, 0, 1) \Rightarrow (0, 1, 1) \Rightarrow (1, 1, 1) \Rightarrow (1, 0, 1) \Rightarrow (0, 0, 1)$. Dans aucun des états de cette boucle il n'existe d'arc sortant permettant de sortir **définitivement** de cette oscillation. C'est donc un régime oscillatoire permanent.

Comme pour le comportement stationnaire, on ne s'intéresse pas aux états transitoires, on utilisera ici encore le langage *Fair-Path CTL* (paragraphe 5.3.1).

Fair-Path CTL pour les comportements oscillatoires Le marqueur biologique pour lequel on traduit le comportement est, comme pour le comportement stationnaire, associé à une variable interne. D'un point de vue formel, dire que la variable oscille revient à dire que la variable v passe une infinité de fois par une valeur minimale x et une infinité de fois par une valeur maximale y , et ce quelque soit le chemin choisis. Cela se traduit en *Fair-Path CTL* par

$A'F(v = x) \wedge A'F(v = y)$. Ensuite pour obtenir la formule finale il faut s'assurer que l'on reste globalement dans cette boucle d'oscillation pour tous les chemins : on ajoute donc $A'G$ pour obtenir

$A'G(A'F(v = x) \wedge A'F(v = y))$.

Cependant si n'est pas une des bornes de la variable v , il faut s'assurer que l'on oscille entre les bornes x et y et pas au-delà. Ces deux contraintes se traduisent respectivement par $A'F(A'G(v \geq x))$ et $A'F(A'G(v \leq y))$. Au final la formule pour décrire l'oscillation entre deux bornes s'écrit :

$A'G(A'F(v = x) \wedge A'F(v = y) \wedge A'F(A'G(v \geq x)) \wedge A'F(A'G(v \leq y)))$.

Définition 5.3.2 (Comportement oscillatoire borné). *Étant donnée une variable interne v et $x, y \in \mathbb{N}$ tel que $0 \leq x < y \leq b_v$, on note $osc_v(x, y)$ la formule *Fair-Path CTL* suivante :*

$$osc_v(x, y) \equiv \left\{ \begin{array}{l} A'G(A'F(v = x) \wedge A'F(v = y)) \\ \wedge A'F(A'G(v \geq x)) \wedge A'F(A'G(v \leq y)) \end{array} \right.$$

Souvent on dispose de la connaissance qu'une variable doit mais sans avoir connaître précisément les bornes de cette oscillation. Nous avons donc créé une troisième notation pour indiquer un comportement oscillatoire non borné. Dans ce cas on testera toutes les oscillations possibles comprises dans le domaine de la variable.

La formule de l'oscillation générale dépend du nombre de niveaux de la variable pour laquelle on décrit le comportement.

Définition 5.3.3 (Comportement oscillatoire). *Étant donnée une variable interne v , on note OSC_v la formule *Fair-Path CTL* suivante :*

$$OSC_v \equiv \bigvee_{x \in [0, b_v - 1]} A'G(A'F(v = x) \wedge A'F(v = x + 1))$$

Il n'est pas difficile de ce convaincre que :

$$OSC_v \Leftrightarrow \bigvee_{0 \leq x < y \leq b_v} osc(x, y) \Leftrightarrow \bigwedge_{x \in [0, b_v]} \neg td_v(x)$$

Il en résulte que td_v et osc_v couvrent tous les comportements asymptotiques possibles puisque au moins un des $osc_v(x, y)$ ou des $td_v(x)$ doit être vrai.

5.3.3 Des motifs à la demande

Il n'en reste pas moins qu'on peut étendre l'expressivité de nos « macros » en *Fair-Path CTL* pour couvrir les connaissances incomplètes qui sont fréquentes en biologie. On peut par exemple indiquer que l'on n'oscillera pas ou que l'on n'aura pas de comportement stationnaire pour une variable particulière. Lorsqu'on utilise la négation pour exclure un motif comportemental, d'un point de vue dynamique, on laisse la possibilité d'avoir plusieurs comportements.

Par exemple si l'on décide d'utiliser le motif $td(v,0)$ où v est une variable dont le domaine est $D_v = \llbracket 0, 2 \rrbracket$, alors v peut avoir un comportement oscillatoire entre 1 et 2 ou tendre vers 1 ou 2.

Les motifs que nous venons de présenter sont très utilisés pour la représentation de systèmes liés au métabolisme. Cependant, dans d'autres systèmes biologiques, lorsqu'on dispose d'informations complémentaires, il ne faut pas se contenter des motifs proposés précédemment, mais compléter cette liste de motifs comme on peut le faire pour le modèle de *Pseudomonas aeruginosa*.

Exemple 5.3.4 – Par exemple les connaissances du comportement d'un système biologique peuvent faire appel à la **non atteignabilité** : « Lorsque une variable v est dans un état x on n'atteindra globalement jamais l'état y ». Cette affirmation peut être formalisée par la formule *Fair-Path CTL* suivante : $NA_v(x, y) \equiv v = x \rightarrow A'G(\neg(v = y))$.

Les motifs et ou formules spécifiquement créés pour représenter les comportements d'un système biologique peuvent être formalisés dans d'autres langages que *Fair-Path CTL* ou *CTL*. Notamment, nous verrons qu'il existe la logique de Hoare dans le chapitre 6, une logique qui permet de mettre en avant des traces comportementales complètes.

Dans cette section qui discute de la manière de formaliser le comportement de chaque marqueur biologique en fonction de l'environnement. Cependant cette description pour l'ensemble du système biologique devient volumineuse lorsque nous avons affaire à de gros modèles. Dans un souci de lisibilité et de traçabilité, il est utile de les répertorier et de les représenter de manière compacte, complète et compréhensible. C'est le propos de la section suivante.

5.4 Formalisation de la matrice de phénotypes

Cette section propose de stocker toutes ces propriétés comportementales dans une «**matrice phénotypique**» où les lignes correspondent aux différents environnements possibles du réseau et les colonnes aux variables internes.

5.4.1 Structure matricielle

Définition 5.4.1 (Matrice de phénotypes). (voir table 8.9) La matrice de phénotype est une matrice

$$MP = (MP_{e,v})_{e \in E, v \in VI},$$

où E est l'ensemble des environnements et VI est l'ensemble des variables internes : Chaque élément $MP_{e,v}$ est une formule *Fair-Path CTL* spécifiant le comportement de la variable interne v lorsque l'environnement est défini sur e .

Une ligne d'une matrice phénotypique fait intuitivement référence à l'observation des différents phénotypes des variables internes dans un environnement particulier.

Définition 5.4.2 (Propriété environnementale d'une matrice phénotypique). La propriété environnementale d'une matrice phénotypique MP pour l'environnement e est la propriété environnementale Φ_e^{MP} suivante :

$$\Phi_e^{MP} \equiv \bigwedge_{v \in VI} MP_{e,v}$$

Les matrices de phénotypes seront principalement utilisées pour représenter les comportements attendus au sein d'un contexte de modélisation. Elles peuvent aussi bien être utilisées lors de l'analyse des besoins que lors de la prédiction des comportements après instantiation des différents contextes de modélisation pour obtenir des prédictions comportementales lisibles.

Pour l'heure ce qui nous intéresse c'est la formalisation des comportements dans le contexte d'analyse des besoins où l'on crée le premier contexte de modélisation. A ce stade, la matrice de phénotype est appelée **matrice de validation**. Cette matrice pourra être creuse : en effet, lorsque les connaissances biologiques ne permettent pas de définir le comportement d'une variable interne dans un environnement particulier, nous n'avons aucune formule à indiquer dans la case correspondante : on laissera la case vide plutôt que d'écrire une tautologie (mais il n'en reste pas moins qu'une case vide « contient » formellement une formule logique toujours vrai).

5.4.2 Illustration avec *Pseudomonas aeruginosa*

En reprenant tous les exemples pour la conception du modèle de *Pseudomonas aeruginosa*, nous pouvons éditer la matrice de validation donnée table 5.1 qui formalise les comportements que l'on cherche à atteindre. On a deux environnements pour *Pseudomonas aeruginosa* définis par les valeurs de la variable d'environnement **Calcium**, e^0 où **Calcium** = 0 et e^1 où **Calcium** = 1.

Environnement	Phénotypes	
Calcium	Operon	MucB
0	$(((\text{Operon} = 2) \Rightarrow A'G((\text{Operon} = 2)))$ $((\text{Operon} < 2) \Rightarrow A'G((\text{Operon} < 2))))$	
1	td(2)	

Table 5.1 – Matrice de validation pour le contexte de modélisation associé à la production de mucus par *Pseudomonas aeruginosa*

Le graphe de régulation possède deux variables internes **Operon** et **MucB**. Nous aurons donc 2 lignes représentant les deux états possibles de **Calcium** dans la matrice et 3 colonnes : une colonne pour l'environnement E réduit ici à l'état du **Calcium**, et deux pour représenter les phénotypes des variables internes (**Operon** et **MucB**). Si l'on veut faire simple, on peut directement mettre les formules données dans l'exemple 5.2.3 dans la case correspondante.

La formule :

$$\varphi^0 \equiv (((\text{Operon} = 2) \Rightarrow A'G((\text{Operon} = 2)))$$

$$((\text{Operon} < 2) \Rightarrow A'G((\text{Operon} < 2))))$$

correspond à l'environnement e^0 et est associée à la variable **Operon** dans la matrice PM . Elle doit figurer dans la case $PM_{e^0, \text{Operon}}$.

De la même manière, pour l'environnement e^1 nous avons formalisé le comportement de **Operon** par la formule suivante : $\varphi^1 \equiv A'G(A'F(\text{Operon} = 2))$. Cette formule correspond au comportement stationnaire traduisible par $td_{\text{Operon}}(2)$. Dans la case $PM_{e^1, \text{Operon}}$ correspondante on pourra donc indiquer td(2), car ici, il n'y a plus besoin de préciser la variable déduite de la colonne dans la matrice.

Malheureusement aucune connaissance biologique n'est apportée concernant le comportement de la variable *MucB*. Dans ce cas les cases qui lui sont associées resteront vides.

Ce chapitre nous a permis de décrire le lien entre connaissances biologiques usuelles et leurs formalisations. Dans la première section nous avons brièvement discuté de la formalisation des régulations, puis nous sommes allés plus dans le détail pour la formalisation des phénotypes (comportements). En effet, ces derniers sont beaucoup plus difficiles à formaliser correctement. Pour simplifier le processus, nous avons mis en place certains motifs de formules temporelles (comportements stationnaires et oscillatoires) qui nous seront particulièrement utiles dans la mise en place du contexte de modélisation traitant des régulations métaboliques. Les motifs proposés ici permettent d'avoir une idée exhaustive des comportements possibles lorsque le système est stable. En effet, si l'on fait la liste de l'ensemble des comportements observables biologiquement après stabilisation du système pour une variable, on constate que les motifs définis sont suffisants pour énumérer tous les comportements possibles. De plus, la matrice de phénotype est une représentation qui permet de regrouper simplement et clairement les phénotypes. L'utilisation des motifs dans la matrice de validation permet de gagner en lisibilité pour travailler avec les biologistes. Cette représentation permet donc d'ouvrir le dialogue et de réduire les erreurs de formalisation que pourrait faire le modélisateur seul. D'un point de vue informatique, la matrice de phénotype a aussi un fort intérêt pour l'automatisation de la validation de modèles. Nous verrons ceci plus en détail dans le chapitre 7.

Cette représentation sous forme de matrice n'est utilisée que depuis quelques années [[Khoodeeram, 2021](#)], elle est actuellement très simple dans son contenu, puisqu'on formalise les comportements des variables internes une à une. On pourrait très simplement imaginer ajouter d'autres types de colonnes. De cette manière on pourrait représenter par exemple le comportement d'ensemble à la fois des variables et des multiplexes. Enfin, on pourrait imaginer ajouter d'autres logiques pour représenter les comportements des variables comme la logique de Hoare, dont on discutera brièvement plus loin dans le manuscrit.

CHAPITRE 6

Méthodes d'identification des paramètres discrets

*Ce chapitre décrit deux méthodes pour identifier les paramètres. La première se réfère directement à la définition de ces paramètres et fait appel à l'expertise des biologistes sur la dynamique locale. Elle nécessite une bonne connaissance des phénomènes biochimiques, et identifie les paramètres de manière manuelle. La seconde repose sur les comportements globaux du modèle et fait appel à la vérification formelle assistée par ordinateur via *TotemBioNet*. *TotemBioNet* énumère les paramétrisations possibles et vérifie pour chacune d'elles si les phénotypes observés biologiquement la dynamique engendré par la paramétrisation est compatible. Enfin comme la seconde méthode nécessite l'utilisation de *TotemBioNet* le chapitre se termine par une description pratique de son fonctionnement.*

6.1	Identification manuelle des paramètres	59
6.1.1	Le principe général	59
6.1.2	L'expérience de pensée	59
6.1.3	Mise en perspective	61
6.2	Identification formelle des paramètres	61
6.2.1	Le principe général	61
6.2.2	La plate-forme <i>TotemBioNet</i>	62
6.2.3	Algorithme global de <i>TotemBioNet</i>	63
6.2.4	<i>TotemBioNet</i> en pratique	63

L'une des difficultés majeures dans la conception d'un modèle est l'identification des paramètres qui en régissent la dynamique. Deux approches principales sont utilisées aujourd'hui pour proposer des valeurs adéquates aux paramètres dans la théorie de René Thomas. La première méthode consiste à identifier un à un chaque paramètre, à la main, en mettant à profit les connaissances des biologistes sur la dynamique locale du système. La seconde méthode repose sur la vérification formelle. En partant de la description formelle des comportements globaux attendus, nous énumérons toutes les paramétrisations possibles et vérifions pour chacune d'entre elles si les phénotypes, c'est à dire les comportements globaux, sont compatibles avec le contexte de modélisation proposé. Cette deuxième approche utilise *TotemBioNet*, un outil permettant l'énumération efficace des paramétrisations ainsi que leur vérification vis à vis des spécifications phénotypiques.

6.1 Identification manuelle des paramètres

La première manière d'obtenir un modèle est d'identifier les paramètres à la main avec l'aide d'un expert du système à modéliser. Cette méthode "bottom-up" utilise des connaissances à une échelle de modélisation de très bas niveau. L'identification des paramètres se fait néanmoins en suivant une méthodologie qui repose sur le principe d'une expérience de pensée.

6.1.1 Le principe général

Suivant la théorie de R.Thomas, on associe à chaque variable plusieurs paramètres (cf. 3). Les paramètres associés à une même variable décrivent des conditions locales différentes liées une même entité biologique : leurs valeurs sont reliées. Pour cette raison, il est plus facile d'identifier de manière groupée les paramètres en se focalisant sur ceux associés à une même variable et de répéter l'identification des différents paramètres variable par variable. Dans certains cas, il existe des paramètres non opérables. Ces derniers n'ont pas à être identifiés car il ne s'appliquent dans aucun des états du graphe de transitions (voir définition 3.2.2 page 27). L'identification des paramètres associés à chaque variable se fait ensuite par le schéma suivant :

1. Pour simplifier l'identification des paramètres d'une variable, on va définir d'abord ses niveaux basal et maximal en commençant par le paramètre qui ne possède aucune ressource puis celui qui possède toutes ses ressources. C'est l'identification des paramètres dans les cas « simples ».
2. Ensuite on identifie les paramètres qui ont des combinaisons de ressources intermédiaires.

L'identification de chaque paramètre nécessite une collaboration profonde entre modélisateurs et biologistes. L'expérience de pensée consiste à se placer dans les conditions locales associées au paramètre considéré, puis d'utiliser l'expertise des biologistes pour définir la valeur du paramètre.

6.1.2 L'expérience de pensée

Le paramètre $K_{v,\omega}$ de la variable v , comme nous l'avons indiqué dans le chapitre 3, contrôle la dynamique de v dans une configuration particulière de régulations effectives (voir définition 3.2.2, page 26). La valeur du paramètre $K_{v,\omega}$ est la valeur vers laquelle va tendre la variable v lorsqu'elle est localement soumise *ad vitam aeternam* aux régulations indiquées par l'ensemble des ressources ω . Pour identifier chaque paramètre un à un on va donc devoir se placer pour chacun d'entre eux dans le contexte local associé en raisonnant sur le Graphe de régulation Local : le graphe local

associé à $K_{v,\omega}$ représente la variable v ainsi que ses multiplexes ressources et ses multiplexes successeurs (multiplexes pour lesquels la variable v est un atome de la formule).

Exemple 6.1.1 – Le graphe de régulation local pour le paramètre $K_{MucB, \text{prod}}$ de l'exemple du graphe de régulation de la production de mucus chez *Pseudomonas aeruginosa* est donné en figure 6.1. Il décrit les conditions de régulation de la variable **MucB** et indique les régulations que la variable **MucB** peut avoir sur ces cibles *via* les multiplexes en aval de la variable elle-même (**free**).

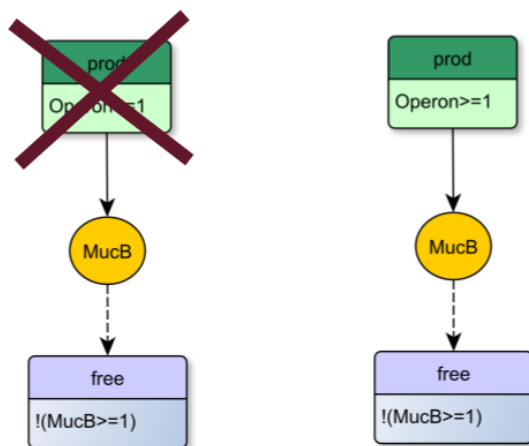


Figure 6.1 – Visualisation des graphes de régulation locaux pour les paramètres K_{MucB} et $K_{MucB, \text{prod}}$ respectivement.

Ensuite, l'expérience de pensée pour le paramètre $K_{v,\omega}$ consiste à définir, à partir du graphe local, tous les multiplexes dont la formule peut être satisfaite après stabilisation en aval de la variable. Pour répondre à cette question on ne se place pas dans un cadre global, mais dans le cadre d'une expérience de pensée où l'on ne considère que le sens des éléments représentés dans le graphe de régulation local. On adopte donc un point de vue conceptuel du graphe et on raisonne avec le biologiste et ses connaissances. Celui-ci nous indique les éléments en aval susceptibles d'être activés.

Une fois que l'on connaît les multiplexes dont la formule doit être satisfaite (ou ne doit pas être satisfaite), cela forme des contraintes de valeurs pour le paramètre considéré. Ces contraintes doivent être résolues pour trouver une valeur du paramètre compatible avec les connaissances biologiques (ou une plage de valeurs).

Exemple 6.1.2 – Lorsqu'on prend le graphe de régulation local associé au paramètre $K_{MucB, \text{prod}}$ donné en figure 6.1 et que l'on raisonne biologiquement, lorsque la production de la protéine MucB se fait, la protéine s'accumule et devient capable d'inhiber l'opéron. Dans le graphe, cette opération se fait *via* le multiplexe **free**. Sa formule doit donc être satisfaite (asymptotiquement) vis à vis des seuils de **MucB**.

Pour que cette régulation soit effective dans les conditions locales considérées, il faut que la formule $(!(MucB \geq 1))$ du multiplexe **free** puisse être évaluée à faux lorsque l'on remplace

Mucb par la valeur du paramètre considéré. Autrement dit il faut que $K_{MucB, \text{prod}} < 1$. Comme $K_{MucB, \text{prod}}$ peut prendre deux valeurs (0 ou 1 car **MucB** est une variable booléenne), il vient $K_{MucB, \text{prod}} = 0$.

Dans cet exemple la formule du multiplexe ne possède qu'un atome dépendant de la variable pour laquelle on cherche à identifier le paramètre par expérience de pensée. Cependant dans des cas plus complexes comme dans le modèle de la régulation du métabolisme, les multiplexes en aval de la variable peuvent contenir une formule plus complexe impliquant plusieurs variables du graphe de régulation. Dans ces cas particuliers, on ne peut prendre que partiellement en compte les atomes des formules n'apparaissant pas dans le graphe local, on les considère comme vérifiés par ailleurs, puisque on s'intéresse à l'impact local d'une seule et même entité.

6.1.3 Mise en perspective

Une fois identifiés, les paramètres forment une paramétrisation partielle pour laquelle on peut éventuellement commencer à vérifier si la dynamique créée est cohérente avec les phénotypes traduits en logique temporelle. Si *TotemBioNet* indique qu'il y a cohérence entre les phénotypes et la paramétrisation partielle, alors on prouve que les régulations indiquées dans le graphe permettent d'obtenir mathématiquement les comportements voulus. Aujourd'hui cette étape peut se faire via *TotemBioNet* (outil de vérification présenté plus tard dans ce chapitre).

Cependant, l'expertise biologique doit être très importante pour réussir à définir tous les paramètres. Ce sont des connaissances de très bas niveau qui sont nécessaires, d'ordre mécanistiques au niveau moléculaire (génétique ou biochimique) et malheureusement elles sont souvent incomplètes et ne permettent pas l'identification de tous les paramètres.

Au vu de la méthode on comprend aisément que pour la conception de réseaux de grande taille, le nombre de paramètres est tel que l'identification manuelle « bottom-up » devient très coûteuse en temps, sinon impossible.

C'est pourquoi cette méthode est complétée ici par la méthode « top-down » : c'est le sujet de la seconde partie de ce chapitre.

6.2 Identification formelle des paramètres

6.2.1 Le principe général

A partir de la description d'un phénotype global que le système doit exhiber et du réseau de régulation associé ainsi que des contraintes issues de l'identification manuelle, le but est de construire l'ensemble de toutes les paramétrisations du réseau qui mènent à une dynamique présentant le comportement souhaité. Pour ce faire le principe est simple. On énumère d'abord toutes les paramétrisations du réseau de régulation restantes. Puis pour chaque paramétrisation P on suivra la définition 3.3.1, page 35 pour savoir si la paramétrisation P donne lieu à un modèle du comportement souhaité formalisé par une formule φ pour le réseau \mathcal{N} . C'est sur ce principe qu'est basé *TotemBioNet*.

6.2.2 La plate-forme *TotemBioNet*

6.2.2.1 Présentation de l'outil

TotemBioNet est un outil dédié à l'identification des paramètres dans les Réseaux de Régulation Biologique [Boyenval et al., 2020]. Étant donné un réseau de régulation et certaines propriétés sur la dynamique du système, *TotemBioNet* génère toutes les paramétrisations et sélectionne celles qui induisent des graphes de transitions pour lesquels les propriétés sont vérifiées. *TotemBioNet* combine les principes de deux outils principaux : *SmBioNet* [Khalis et al., 2009] et Hoare-Fol [Folschette, 2019b]. Pour mieux comprendre le fonctionnement de *TotemBioNet* nous décrivons rapidement ces deux outils puis le Workflow de *TotemBioNet*.

6.2.2.2 *SmBioNet*

SmBioNet était une plate-forme logicielle basée sur la sémantique de R. Thomas. L'outil reposait sur des algorithmes de model checking inspirés des techniques de vérification de programmes [Clarke and Emerson, 1982, Batt et al., 2004, Naldi et al., 2018]. La logique temporelle *CTL* permettait de décrire des propriétés très générales et le model checking vérifiait si ces propriétés sont satisfaites par un graphe de transitions. *SmBioNet* traitait exhaustivement l'ensemble de toutes les paramétrisations par des algorithmes d'énumération.

TotemBioNet reprend la structure pour l'énumération des paramètres de *SmBioNet*, et a recours au même model checker : NuSMV [Cimatti et al., 2002]. Néanmoins, l'énumération de toutes les paramétrisations possibles demande un temps exponentiel car le nombre de paramètres dépend exponentiellement des nombres de prédécesseurs de chacune des variables (cf formule 3.1, page 28). C'est pourquoi *TotemBioNet* intègre aussi les principes de la logique de Hoare modifiée tels que présentés dans l'outil Hoare-fol, qui permet, si l'on connaît une trace précise, la réduction du nombre de paramétrisations par la production d'un ensemble de contraintes sur les paramètres.

6.2.2.3 Hoare-Fol

Hoare-fol [Folschette, 2019a] est un outil développé par Maxime Folschette basé sur la logique de Hoare. La logique de Hoare originelle [Hoare, 1969] permet de raisonner sur la correction des programmes informatiques impératifs. Par l'intermédiaire d'un triplet de Hoare on décrit comment une portion de code fait évoluer l'état de la machine sur laquelle est exécuté le programme. Ce triplet, noté $\{P\}C\{Q\}$, décrit l'état initial de la machine avant l'exécution de la portion du code ($\{P\}$ pour pré-condition), le code impératif (C), et l'état de la machine après exécution du code C ($\{Q\}$ appelé post-condition).

Cette logique de Hoare a été adaptée pour raisonner sur un réseau de régulation génétique [Bernot et al., 2019]. Le programme est remplacé par une trace comportementale que le modèle doit pouvoir suivre. Comme pour la vérification de programmes, il est possible de calculer la plus faible pré-condition, sur les paramètres du réseau de régulation pour que la trace comportementale soit exhibée par le modèle. Autrement dit, au lieu d'énumérer les paramétrisations comme dans l'approche *SmBioNet*, Hoare-fol produit un ensemble de contraintes sur les valeurs des paramètres qui caractérise les valeurs pour lesquelles ces traces existent. Chaque triplet de Hoare dans ce contexte décrit l'état initial, la trace comportementale (grâce à des incréments)

et décréments des variables) ainsi que l'état final.

Biologiquement, ce type de connaissances est rarement aussi détaillé. Plus le système est grand et intriqué moins la description des comportements *via* ces triplets est possible. C'est pourquoi, comme nous nous intéressons à des systèmes biologiques de grande taille, nous n'utilisons pas cette logique pour la vérification formelle des comportements et les contraintes sur les paramètres que nous considérons sont issues de l'identification manuelle décrite précédemment.

L'avancée majeure de *TotemBioNet* est d'être capable d'utiliser séparément ou en coopération à la fois la logique temporelle (*CTL* et/ou *Fair-Path CTL*) et la logique de Hoare génétiquement modifiée ou tout jeu de contraintes obtenu par ailleurs sur les paramètres. Les différentes étapes du fonctionnement de *TotemBioNet* sont décrites dans la sous section suivante.

6.2.3 Algorithme global de *TotemBioNet*

6.2.3.1 Présentation du Workflow

TotemBioNet prend en entrée à la fois un graphe de régulation et des propriétés environnementales écrites en *CTL* et/ou en *Fair-Path CTL* et/ou en logique de Hoare (encadrés gris dans la figure 6.2). L'outil commence par construire des contraintes déduites par la logique de Hoare à partir de la propriété exprimée sous forme de triplet de Hoare, avant d'énumérer toutes les paramétrisations possibles. Si les contraintes ne sont pas disjonctives, ces contraintes permettent de réduire le nombre de paramétrisations à énumérer. Ensuite, pour chaque paramétrisation *TotemBioNet* vérifie si les contraintes disjonctives sont respectées. Si ce n'est pas le cas, la paramétrisation considérée n'est pas satisfaisante. À l'inverse, si les contraintes disjonctives sont respectées, *TotemBioNet* procède, *via* NuSMV [Cimatti et al., 2002], à la vérification formelle de la cohérence entre les formules temporelles données (*CTL* et/ou *Fair-Path CTL*) et les graphes de transitions décrits par le réseau de régulation et la paramétrisation testée. L'outil donne en sortie exhaustivement toutes les paramétrisations du réseau de régulation donné qui sont compatibles avec les propriétés indiquées et les contraintes initiales.

6.2.4 *TotemBioNet* en pratique

En pratique l'outil prend en entrée un fichier au format « .smb » qui contient différents blocs décrivant le graphe (VAR et REG), les paramètres (PARA) si certains ont été préalablement contraints par les connaissances *ad hoc* et enfin les propriétés biologiques (blocs HOARE, CTL ou FAIRCTL).

Exemple 6.2.1 – La figure 6.3 correspond à un fichier d'entrée de *TotemBioNet* pour le modèle de *Pseudomonas aeruginosa* (cf 3.1, page 24). Le bloc VAR décrit les variables (*Operon*, *MucB*, *Calcium*) du graphe de régulation ainsi que leurs domaines de valeurs. Le bloc REG décrit l'ensemble des régulations du graphe *via* la description des multiplexes et de leurs formules. Le bloc PARA n'indique que les paramètres déjà identifiés : ici il n'y en a pas. Le bloc HOARE comprend un triplet de Hoare décrivant une trace observée ou souhaitée, le bloc CTL permet de

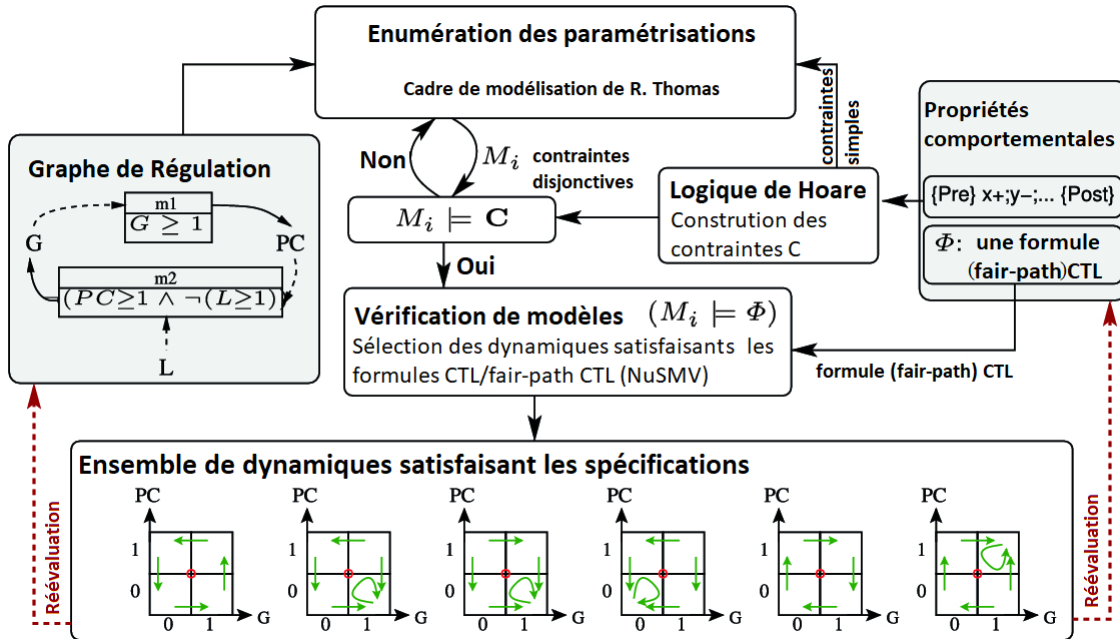


Figure 6.2 – Workflow de *TotemBioNet* traduit et issu de l'article [Boyenval et al., 2020].

spécifier les comportements attendus *via* la logique *CTL*. Enfin si on le souhaite, on peut avoir un bloc FAIRCTL qui décrit en *Fair-Path CTL* les comportements¹.

En réponse, *TotemBioNet* indique le nombre total de modèles testés, le nombre de modèles sélectionnés et le temps de calcul.

Pour cet exemple, le nombre de paramétrisations possible est de 504², le nombre de paramétrisations qui valident les spécifications sont au nombre de 36 et sont sélectionnées en 3s508ms³.

Enfin notons que d'un point de vue pratique *TotemBioNet* aussi est capable de gérer les variables d'environnements à l'aide du bloc ENVAR.

Dans ce chapitre nous avons décrit les expériences de pensée permettant d'identifier manuellement les paramètres dynamiques d'un réseau de régulation. Cette technique est complétée ensuite par de la vérification formelle *via* l'outil *TotemBioNet*. *TotemBioNet* vérifie si les comportements traduits en formules logiques (*CTL Fair-Path CTL* ou triplés de HOARE) sont cohérents avec les différentes paramétrisations possibles du réseau. Une fois vérifiées formellement, chacune des paramétrisations cohérentes avec les spécifications comportementales constitue avec le réseau associé un modèle.

1. Si les formules sont indiquées en *Fair-Path CTL*, *TotemBioNet* se charge de les convertir en *CTL* pour continuer d'utiliser le model checker NuSMV qui n'accepte que des formules *CTL*.

2. Par défaut *TotemBioNet* prend en compte les conditions de Snoussi : l'estimation du nombre de paramétrisations réel n'est pas un calcul trivial, raison pour laquelle le nombre indiqué est celui calculé par *TotemBioNet*.

3. Intel® Core™ i7-7700 CPU/3.60 GHz × 8, RAM : 32 Go, sous Linux


```

VAR
operon = 0 .. 2 ;
mucuB = 0 .. 1 ;
calcium = 0..1;

REG
prod [(operon>=1)] => mucuB;
free [!(mucuB>=1)] => operon;
alg [(operon>=2)] => operon;
ca [calcium>=1] => operon;

# Hoare triple
HOARE

    PRE : {mucuB=0, operon=1}
    TRACE : mucuB+; operon-;
    POST : {mucuB=1, operon=0}

CTL
# steady states for operon
((operon = 0) -> AG(!(operon = 2)));
((operon = 2) -> AG(!(operon = 0)));

FAIRCTL
AF (AG (mucuB>=1));

END

```

Figure 6.3 – Fichier d’entrée de *TotemBioNet* pour le modèle de *Pseudomonas aeruginosa* avec logique de Hoare, *CTL* et *Fair-Path CTL*.

Le prochain chapitre adapte l’utilisation de *TotemBioNet* et le principe de vérification formelle en proposant deux nouvelles méthodes de validation de modèles qui prennent en compte la notion d’environnements et de matrice de phénotype (introduites en chapitre 5).

CHAPITRE 7

Méthodes de vérification de modèles

La vérification formelle des propriétés d'une matrice de comportement est une tâche assez complexe puisque les propriétés dépendent des variables d'environnement. Deux approches sont proposées dans ce chapitre.

La première repose sur la construction d'un réseau de régulation unique qui représente les régulations dans chacun des environnements. Cette approche globale, permet d'obtenir l'ensemble des paramétrisations cohérentes en un seul lancement de *TotemBioNet*.

La seconde est basée sur une approche environnementale : pour chaque environnement on construit un réseau de régulation qui représente le comportement dans cet environnement, puis on lance *TotemBioNet* pour cet environnement. Cette approche nécessite une dernière étape : combiner les modèles obtenus pour chacun des environnements.

7.1	Méthode de validation globale de la matrice	68
7.1.1	Autorégulations pour simuler la stabilité d'un environnement	69
7.1.2	Séparation des plans via les artefacts chez <i>Pseudomonas aeruginosa</i>	70
7.1.3	Formules pour l'ensemble des environnements	71
7.1.4	Vérification globale via <i>TotemBioNet</i>	72
7.2	Formalisation de la méthode par environnements	73
7.2.1	Un réseau de régulation avec environnement	73
7.2.2	Formules et abstractions de modèles	75
7.2.3	Équivalence des approches	77
7.2.4	Validation environnement par environnement via <i>TotemBioNet</i>	78
7.3	Automatisation de la méthode environnement par environnement	80
7.3.1	Utilisation de la matrice de validation	80
7.3.2	Automatisation par <i>GreenBioNet</i>	80
7.3.3	Exemple	83
7.4	Conclusion	83

Pour les petits réseaux de régulation, la validation peut généralement se faire en une seule étape car les systèmes décrits sont pour la plupart développés pour étudier leurs comportements dans un seul environnement et ce de manière implicite. En évoluant vers la conception de plus gros systèmes, on a vu apparaître la notion d'environnements et de phénotypes associés à chacun d'entre eux pour un même système biologique, comme nous l'avons expliqué au chapitre 5. Il a donc fallu adapter notre démarche pour garantir la validation des modèles. C'est pourquoi ce chapitre propose deux méthodes de validation adaptées aux réseaux dont le comportement dépend des environnements [Gibart et al., 2021b].

La première méthode permet l'étude de situations environnementales successives dans un réseau global unique (cf. Figure 7.1). Cette méthode permet la recherche des paramétrisations compatibles avec les connaissances biologiques pour tous les environnements. Ceci est rendu possible par l'ajout d'autorégulations sur les variables d'environnements. Enfin, *TotemBioNet* permettra alors de vérifier, en une passe, l'ensemble des propriétés comportementales des différents environnements.

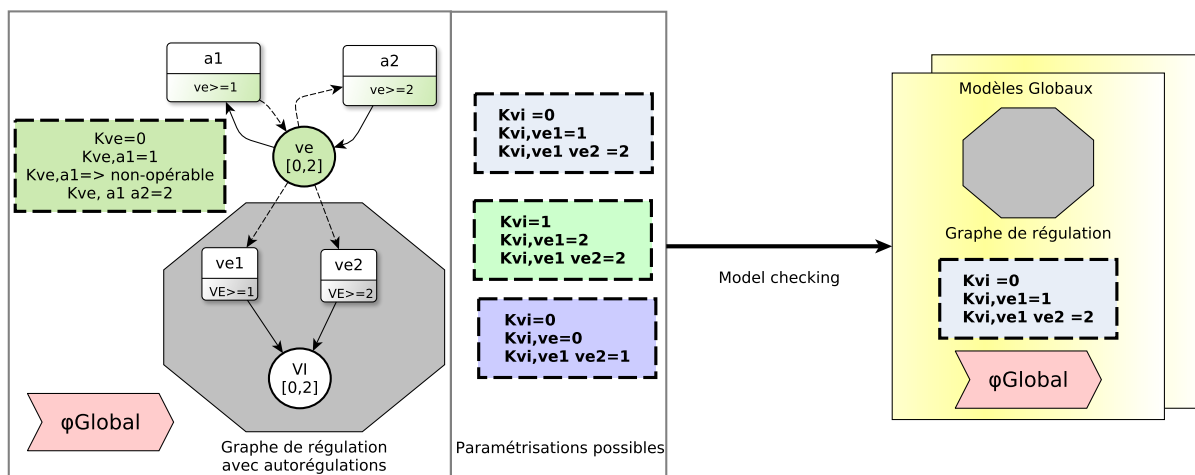


Figure 7.1 – Méthodologie globale. Les autorégulations sur les variables d'environnement permettent alors de fixer les variables d'environnements, et ainsi de simuler les comportements différents pour chacun des environnements.

La seconde méthode est présentée figure 7.2. Elle se base sur une approche « diviser pour régner » en utilisant la notion d'environnements et de phénotypes que nous avons décrits au chapitre 5. Pour chaque environnement, le réseau de régulation est adapté et des paramétrisations compatibles avec les connaissances biologiques associées sont recherchées dans un espace de recherche beaucoup plus petit que dans la première méthode. Ensuite, ces ensembles de paramétrisations sont regroupés pour obtenir ceux qui donnent les traces cohérentes avec toutes les observations des différents environnements. Cette approche « diviser pour régner » est plus efficace que l'approche globale.

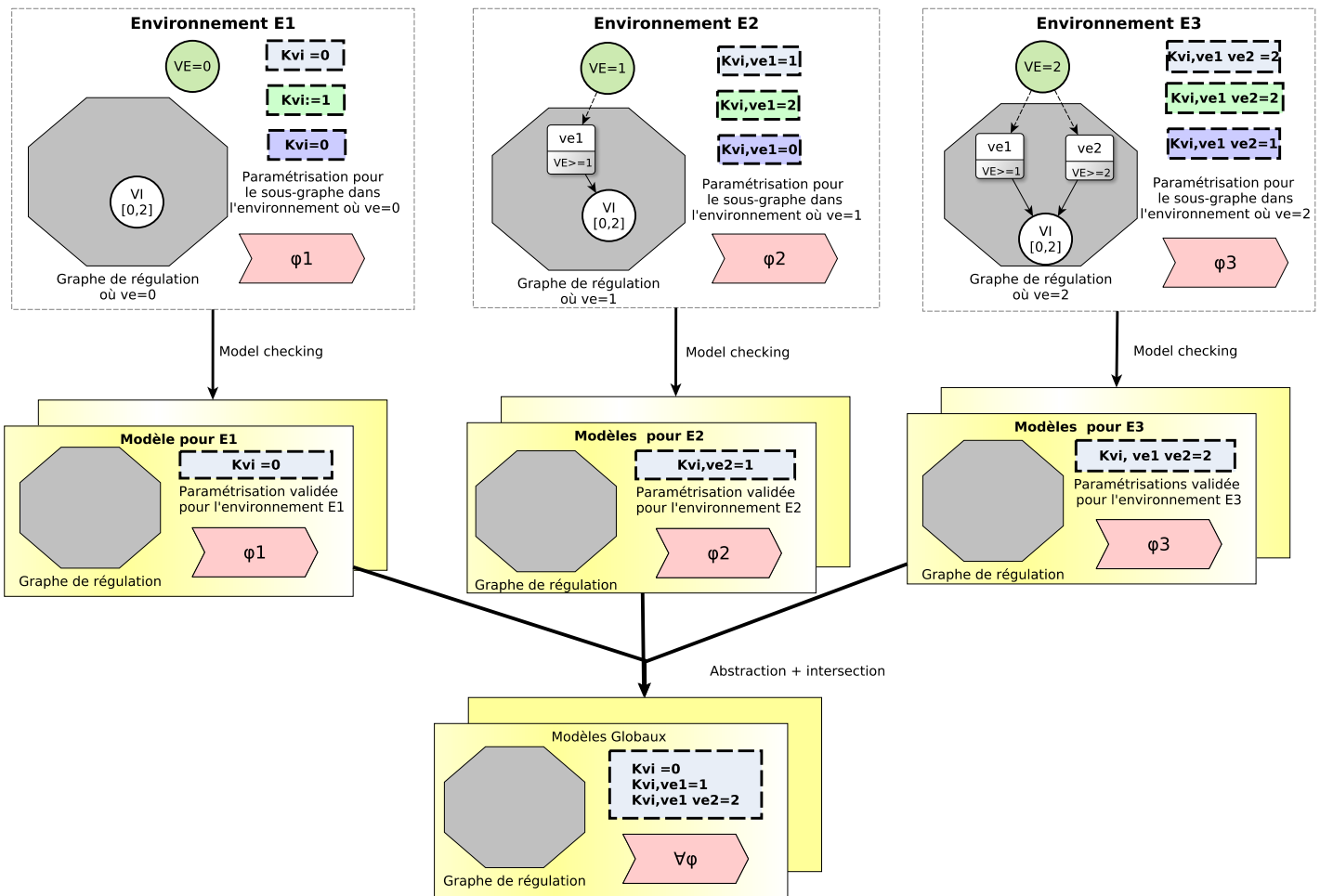


Figure 7.2 – Schéma des étapes de l'approche diviser pour régner. Chaque environnement est étudié séparément, menant à un ensemble de modèles cohérents spécifiques. La dernière étape consiste à construire les modèles globaux à partir des modèles cohérents aux différents environnements.

7.1 Méthode de validation globale de la matrice

La présence d'une variable externe peut drastiquement modifier le comportement des variables internes, et lorsque il y a plusieurs variables d'environnement, chaque combinaison de présence/absence des variables externes peut entraîner une dynamique différente. Si l'on veut vérifier pour une même paramétrisation des propriétés portant sur des environnements différents, il est donc nécessaire d'avoir un réseau qui permette de simuler les comportements dans chacun des environnements. Or, si l'on se réfère à la description d'un graphe de régulation 3.1.1, page 23 (chapitre 3), les variables d'environnement n'ont aucun prédécesseur. Cela implique que chaque variable d'environnement n'a qu'un seul paramètre. Par défaut, ce paramètre provoque une transition constante d'un plan vers un autre dans le graphe de transitions.

Exemple 7.1.1 – Reprenons l'exemple du modèle de *Pseudomonas aeruginosa* donné au chapitre 3. Le tableau de paramètres de la table 3.3 donne $K_{\text{Calcium}} = 1$, il n'y a qu'un seul paramètre pour le calcium, et sur le graphe de transition associé donné figure 3.5, on peut vérifier qu'il y a des transitions qui partent du plan où $\text{Calcium}=0$ vers le plan où $\text{Calcium}=1$.

On comprend donc que même en fixant les valeurs des variables d'environnement, le graphe de régulation et les paramètres seuls ne nous permettent pas de tester des comportements différents dans des environnements différents.

7.1.1 Autorégulations pour simuler la stabilité d'un environnement

Pour dépasser cette limitation, il faut avoir la possibilité de « simuler » dans le même réseau de régulation les différents environnements. Une des solutions est de rajouter des autorégulations sur les variables d'environnement dont le seul rôle est de stabiliser la valeur des variables d'environnements artificiellement. Ainsi, chaque environnement est représenté dans le graphe de transition par un hyper-plan. Ajouter ces autorégulations permet de rendre ces plans indépendants : les paramètres associés aux variables d'environnement vont permettre de séparer les hyper-plans.

7.1.1.1 Les autorégulations

Supposons que la variable d'environnement $v \in VE$ prend sa valeur dans $\llbracket 0, b_v \rrbracket$. Alors, pour chaque valeur $n \in \llbracket 1, b_v \rrbracket$, on ajoute un multiplexe A_v^n de formule $v \geq n$. Ainsi, lorsque $v < n$ (resp. $v \geq n$), le multiplexe A_v^n n'est pas (resp. est) une ressource de v . Les multiplexes A_v^n jouent un rôle particulier dans cette formalisation, car ils ne représentent aucun aspect particulier du système biologique, ils permettent au *GTA* de simuler l'existence de différentes valeurs stables pour les variables d'environnement. Dans ce sens, ils sont *artéfactuels*.

Exemple 7.1.2 – Un exemple d'autorégulation pour la validation globale est donné figure 7.1. La variable v prend sa valeur dans $\llbracket 0, 2 \rrbracket$, 2 autorégulations sont ajoutées : **a1** de formule $v \geq 1$ et **a2** de formule $v \geq 2$.

7.1.1.2 La stabilité des autorégulations

Puisque v est une variable d'environnement, les seuls prédécesseurs sont les multiplexes artéfactuels A_v^k , $k \in \llbracket 1, b_v \rrbracket$. Comme nous avons construit autant de multiplexes artéfactuels que de valeurs possibles pour la variable, il y a 2^{b_v} paramètres (cf. équation 3.1 du chapitre 3). Heureusement, certains d'entre eux sont structurellement non applicables (inutiles pour déterminer la dynamique, voir Def. 3.2.2) et n'ont pas à être instanciés : en effet lorsque $v = k$ ($k \in \llbracket 0, b_v \rrbracket$), les seules ressources de v sont les multiplexes A_v^i avec $i \leq k$. Ainsi les paramètres $K_{v,\omega}$ tels que $A_v^i \in \omega$ et $A_v^j \notin \omega$ avec $j < i$ ne sont pas applicables. Finalement, quand $v \in VE$ peut prendre sa valeur dans $\llbracket 0, b_v \rrbracket$, il existe exactement $b_v - 1$ paramètres applicables : $K_v, K_{v,\{A_v^1\}}, K_{v,\{A_v^1, A_v^2\}}, \dots, K_{v,\{A_v^1, A_v^2, \dots, A_v^{b_v}\}}$.

Exemple 7.1.3 – Toujours en utilisant la figure 7.1, avec les autorégulations artéfactuelles, les paramètres possibles pour la variable v sont : $K_v, K_{v, \mathbf{a1}}, K_{v, \mathbf{a2}}$ et $K_{v, \mathbf{a1 a2}}$. Le paramètre $K_{v, \mathbf{a2}}$

n'est pas applicable : la variable ne peut pas à la fois être supérieure à son second seuil (**a2** pour ressource si $v \geq 2$) et être inférieure à son premier seuil (**a1** n'est pas ressource si $v \geq 1$). Ce paramètre ne sera donc jamais applicable.

Pour stabiliser les différents environnements, on va donner des valeurs aux paramètres applicables de v . Comme on souhaite ne pas pouvoir sortir de chaque environnement, la valeur de la variable v ne doit pas être modifiable. Lorsque $v = 0$ le paramètre qui s'applique est K_v . Par conséquent K_v doit être égal à 0. De même lorsque $v = 1$, on veut que la valeur de la variable v soit stable donc le paramètre qui s'applique $K_{v, \mathbf{a1}}$ doit valoir 1. Enfin, lorsque $v = 2$ le paramètre applicable $K_{v, \mathbf{a1 a2}}$ doit valoir 2.

Cette démarche est généralisable à tout graphe de régulation. Pour s'assurer que les variables d'environnement ne soient pas attirées vers une valeur différente de leur valeur initiale, il faut imposer des valeurs aux paramètres applicables. Lorsque la variable d'environnement v vaut k ($k \in \llbracket 0, b_v \rrbracket$), la valeur du paramètre effectif doit aussi être k . Ainsi $K_v = 0$, $K_{v, \{A_v^1\}} = 1$, $K_{v, \{A_v^1, A_v^2\}} = 2, \dots, K_{v, \{A_v^1, A_v^2, \dots, A_v^{b_v}\}} = b_v$.

7.1.2 Séparation des plans via les artéfacts chez *Pseudomonas aeruginosa*

Exemple 7.1.4 – Reprenons l'exemple du réseau de régulation de *Pseudomonas aeruginosa*. La seule variable d'environnement est **Calcium**. Sachant que son niveau maximal est 1 on ne construit qu'un seul multiplexe autorégulateur, qui aura pour formule **Calcium** ≥ 1 (cf. Figure 7.3 en vert). Les paramètres associés à la variable **Calcium** apparaissent aussi dans la figure, encadrés en pointillés sur fond vert.

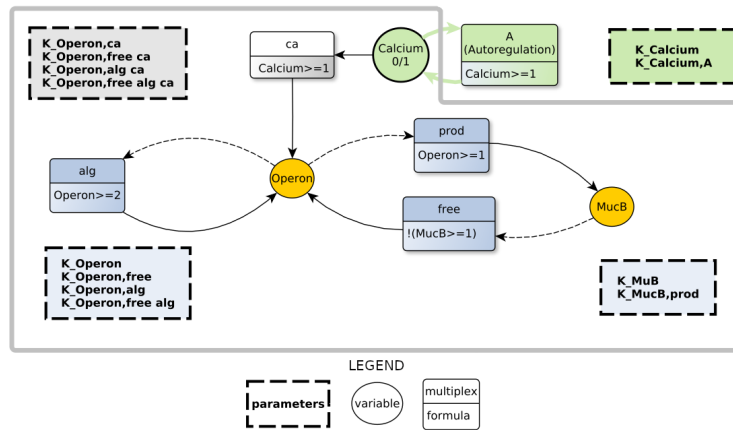


Figure 7.3 – réseau de régulation du modèle de production de mucus par *Pseudomonas aeruginosa* avec autorégulation sur la variable **Calcium** (indiquée en vert). Les paramètres sont indiqués en syntaxe concrète de *TotemBioNet*.

Avec $K_{calcium} = 0$ et $K_{Calcium, \mathbf{A}} = 1$ et les valeurs de tous les autres paramètres ($K_{Operon, \mathbf{ca}} = 0$, $K_{Operon} = 0$, $K_{Operon, \mathbf{free ca}} = 1$, $K_{Operon, \mathbf{free}} = 1$, $K_{Operon, \mathbf{alg ca}} = 1$, $K_{Operon, \mathbf{alg}} = 1$, $K_{Operon, \mathbf{alg free ca}} = 2$, $K_{Operon, \mathbf{alg free}} = 2$, $K_{MucB} = 0$, $K_{MucB, \mathbf{prod}} =$

1, $K_{\text{Calcium}} = 1$ ($K_{\text{Calcium}, \mathbf{A}} = 1$), nous pouvons construire le graphe de transition associé (cf. Figure 7.4).

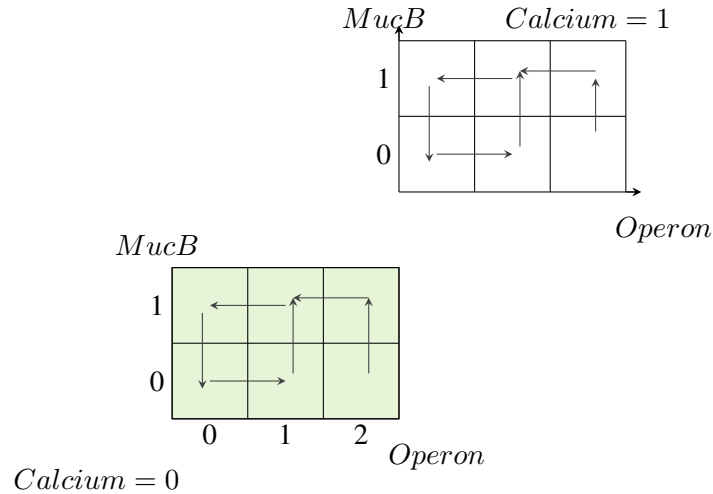


Figure 7.4 – Graphe de transitions pour *Pseudomonas aeruginosa* avec l'autorégulation pour simuler les environnements de **Calcium**. En vert les états où **Calcium=0**.

Lorsque **Calcium** = 0 (états en fond vert dans le graphe de transitions Figure 7.4) le paramètre qui s'applique à **Calcium** est K_{calcium} , le paramètre valant 0 il n'existe aucune transition possible allant du plan où **Calcium=0** vers le plan où **Calcium=1**. Lorsque **Calcium** = 1, le paramètre qui s'applique à **Calcium** est $K_{\text{Calcium}, \mathbf{A}}$ et comme sa valeur est 1 on ne pourra pas transiter du plan où **Calcium=1** vers le plan où **Calcium=0** : les plans sont donc « stables ».

7.1.3 Formules pour l'ensemble des environnements

Dans le chapitre 5 nous avons indiqué en définition 5.2.2 que nous pouvons exprimer les propriétés comportementales en fonction de l'environnement à l'aide du langage *CTL*. Une propriété dans un environnement est représentée par un couple (e, φ) .

Définition 7.1.1 (Propriété biologique dans un environnement). *Le couple (e, φ) défini*

Nous pouvons donc représenter l'ensemble des comportements de la matrice de validation par une liste de propriétés environnementales $[(e^i, \varphi^i)]_{i=1}^n$. Pour traduire ces propriétés environnementales en formules *CTL* il est nécessaire de caractériser les états de chacun des environnements. La caractérisation des états de l'environnement e^i est donnée par la formule :

$\varepsilon^i \equiv \bigwedge_{v_k \in VE} (v_k = e_k^i)$ où e_k^i représente la valeur de la variable d'environnement v_k dans l'environnement e^i . Une fois l'environnement traduit, il reste à l'associer à la formule correspondante. On traduira une propriété φ^i dans un environnement e^i par la formule : $\varepsilon^i \Rightarrow \varphi^i$. Ainsi, lorsque

on souhaite une liste de propriétés environnementales $[(e^i, \varphi^i)]_{i=1}^n$ on écrira la formule :

$$\Phi_{global} \equiv \bigwedge_{i \in \llbracket 1, n \rrbracket} (\varepsilon^i \Rightarrow \varphi^i)$$

où ε^i caractérise les états de l'environnement e^i .

L'utilisation de l'implication pour définir l'environnement dans lequel on va tester les différents comportements permet de tester toutes les propriétés d'environnement *via* le model-checking de *TotemBioNet*. A partir du réseau de régulation modifié et de la formule Φ_{global} , *TotemBioNet* énumère toutes les paramétrisations et ne sélectionne que celles qui sont cohérentes avec Φ_{global} .

Exemple 7.1.5 – Le modèle de *Pseudomonas aeruginosa* est défini à partir de φ^0 et φ^1 que l'on a déjà explicitées chapitre 5 en sous-section. 5.2.3 et 3.3.1 :

$$\begin{aligned} \varphi^0 : & \quad (((Operon = 2) \Rightarrow A'G((Operon = 2))) \ \& \\ & \quad ((Operon < 2) \Rightarrow A'G((Operon < 2)))) \\ \varphi^1 : & \quad A'G(A'F(Operon = 2)) \end{aligned}$$

En ajoutant la caractérisation des environnements pour chaque formule on obtient la formule Φ_{global} :

$$\Phi_{global} \equiv ((Calcium = 0) \Rightarrow \varphi^0) \ \wedge \ ((Calcium = 1) \Rightarrow \varphi^1)$$

Φ_{global} sera ensuite utilisé telle quelle pour la vérification globale *via* *TotemBioNet*.

7.1.4 Vérification globale *via* *TotemBioNet*

D'un point de vue pratique, rechercher des paramétrisations cohérentes avec ϕ_{global} se fait par l'intermédiaire d'une seule commande et d'un seul fichier d'input avec *TotemBioNet*. Dans ce fichier figurent le graphe de régulation avec les artefacts, les paramètres en partie identifiés (comprenant les paramètres valués associés aux artefacts) et les propriétés en fonction de l'environnement *via* Φ_{global} . *TotemBioNet* va énumérer l'ensemble des paramétrisations possibles pour le réseau de régulation donné \mathcal{N} .

Ensuite *TotemBioNet* effectue la vérification de l'ensemble des paramétrisations *via* le model checking $check_{\mathcal{N}}$ défini au chapitre 3 (notation 3.3.1, page 35).

Exemple 7.1.6 – La validation de *Pseudomonas aeruginosa* par l'approche globale. Supposons que nous n'ayons au préalable identifié aucun des paramètres exceptés ceux associés à l'artefact (les paramètres sont indiqués en figure 7.3).

Operon a 3 prédécesseurs (**alg**, **free** et **ca**), donc 2^3 paramètres et peut prendre 3 valeurs différentes (**Operon** $\in \llbracket 0, 2 \rrbracket$). Le nombre de paramétrisations pour **Operon** est donc de $3^{2^3} = 6561$. **MucB** a 1 prédécesseur unique (**prod**), soit 2^1 paramètres et peut prendre 2 valeurs différentes (**MucB** $\in \llbracket 0, 1 \rrbracket$) : le nombre de paramétrisations de **MucB** est donc de $2^{2^1} = 4$. Ainsi, le nombre de paramétrisations à prendre en compte est : $6561 \times 4 = 26244$. Par l'approche globale ces 26 244 jeux de paramètres sont énumérés et vérifiés par rapport à la formule globale Φ_{global} en 147,238 secondes sur un ordinateur personnel¹.

1. Les temps d'exécutions sont donnés sur une moyenne de 20 lancements de *TotemBioNet* sur un ordinateur Intel® coeur™ i7-7700 CPU/3.60GHz \times 8, RAM : 32 Go, sous Linux. Le fichier d'entrée et les résultats sont disponibles

Dans cet exemple nous avons décrit le calcul du nombre de paramétrisations intuitivement en calculant le nombre de paramètres possibles pour chaque variable. Formellement le nombre de paramétrisations est le suivant :

$$\prod_{v \in V} (b_v + 1)^{(2^{i_v})} \quad (7.1)$$

où i_v le degré entrant de la variable v (nombre de régulations) et b_v la valeur maximale du domaine $\llbracket 0, b_v \rrbracket$ associé à v . Le terme $b_v + 1$ correspond donc au nombre de valeurs que peut prendre la variable v dont le domaine est $\llbracket 0, b_v \rrbracket$.

Passons maintenant à la description de la seconde approche, qui se base sur une vérification environnement par environnement.

7.2 Formalisation de la méthode par environnements

La démarche de notre seconde approche « diviser pour régner » (cf. Figure 7.2.) se base sur l'utilisation des différents environnements. En se plaçant dans un environnement on se focalise sur la sous-partie utile (pour l'environnement choisi) du réseau de régulation. Puis on vérifie, pour chacune de ces sous-parties, les propriétés environnementales associées. Enfin, après avoir déduit toutes les paramétrisations acceptables pour chacune de ces sous-parties, il convient de reconstruire l'ensemble exhaustif des paramétrisations du réseau global cohérent avec toutes les propriétés environnementales. Cette opération consiste principalement à abstraire puis à croiser les modèles trouvés dans chaque environnement spécifique.

7.2.1 Un réseau de régulation avec environnement

Au sein du graphe de régulation où l'on considère les environnements (définition 5.1.1, page 49), donner une valeur à chaque variable d'environnement réduit le nombre de régulations effectives. Sachant que le nombre de paramétrisations est calculé à l'aide de l'équation donnée 7.1, le nombre de paramétrisation est drastiquement réduit par rapport à l'approche globale, car le nombre d'entrée pour toutes les variables internes directement liées aux variables d'environnements est réduit. De plus l'espace d'états considéré sera réduit à l'hyperplan défini par la valeur choisie pour les variables d'environnement.

Définition 7.2.1 (Espace d'états pour un environnement). *Étant donné un graphe de régulation avec variables d'environnement $G = (V, VE, M, A)$ et un environnement e , l'espace d'états du système pour e est $S_e = \prod_{v \notin VE} \llbracket 0, b_v \rrbracket \times \prod_{v_k \in VE} \{e_k\}$.*

Exemple 7.2.1 – Le nombre d'états d'un graphe de transitions global de *Pseudomonas aeruginosa* est de 12 comme nous l'avons indiqué exemple 3.2.1. Maintenant plaçons nous dans le cas où le

en suivant le lien : <https://gitlab.com/totembionet/totembionet/-/tree/master/examples/CMSB2021>.

calcium prend la valeur 0 (nous sommes dans l'environnement e^0); le nombre d'états associés est donc de $3 \times 2 \times 1 = 6$ (cf définition 3.2.1). Le nombre d'états a été divisé par 2. La diminution du nombre d'états est exponentielle en fonction du nombre de variables d'environnement.

Ainsi, pour un environnement, le nombre d'états du graphe de transitions dans un environnement est forcément plus petit que le nombre d'états du graphe de transition initial (donné par la définition 3.2.1 dans le chapitre 3). Se placer dans un environnement revient donc à se placer dans un sous ensemble d'états S_e de S .

Exemple 7.2.2 – Considérons que l'on se place dans l'environnement où **Calcium** = 0 (e^0), les états où **Calcium=1** sont par définition non atteignables. Si ces états sont non atteignables, l'ensemble des paramètres dans ces états ne s'appliqueront jamais dans cet environnement : ils seront donc non-opérables pour l'environnement. Ainsi, les paramètres de **Operon** qui ont **ca** pour ressource ne seront pas opérables dans e^0 (correspond aux paramètres indiqués en fond gris dans la figure 7.3 : K_{Operon} , $K_{Operon, \mathbf{alg}}$, $K_{Operon, \mathbf{free}}$, $K_{Operon, \mathbf{alg free}}$). A l'inverse lorsque l'on sera dans l'environnement e^1 , les paramètres dans les rectangles en pointillés sur fond gris seront les seuls qui serait opérables.

Définition 7.2.2 (Paramètres opérables pour un environnement). *Étant donné G et un environnement e , un paramètre $K_{v,\omega}$ est opérable dans l'environnement e si il existe au moins un état $s \in S_e$ dans lequel $K_{v,\omega}$ est applicable (Cf. notion d'applicabilité chapitre 3 définition 3.2.2).*

Au final, si l'on considère un graphe de régulation pour un environnement et ses paramètres associés on construit un réseau de régulation pour un environnement qui sera défini comme suit :

Définition 7.2.3 (Réseau de régulation dans un environnement). *Un réseau de régulation avec environnement $e \in E$ est le couple $\mathcal{N}_e = (G, \mathcal{K}_e)$ où $\mathcal{K}_e \subset \mathcal{K}$ est le sous-ensemble des paramètres opérables pour e . Un paramétrage P_e attribue à chaque $K_v \in \mathcal{K}_e$ une valeur dans $\llbracket 0, b_v \rrbracket$. L'ensemble de toutes les paramétrisations pour e est noté $P_{\mathcal{N}_e}$.*

Définition 7.2.4 (GTA pour une paramétrisation P_e dans l'environnement e). *Étant donné $G_e = (G, \mathcal{K}_e)$ un réseau de régulation pour l'environnement e et $P_e \in P_{\mathcal{N}_e}$ une paramétrisation pour \mathcal{N}_e , le GTA_{P_e} associé est défini comme suit :*

- **Noeuds** : sont des états $s \in S_e$,
- **Arrêtes** : il existe un arc de $s^p = (s_1^p, \dots, s_n^p)$ à $s^q = (s_1^q, \dots, s_n^q)$ s'il existe un et un seul indice i s.t. $v_i \in V \setminus VE$, et $s_i^p \neq s_i^q$ tel que :
 - soit $s_i^q = s_i^p + 1$ et $P_e(K_{v_i}[s^p]) > s_i^p$
 - soit $s_i^q = s_i^p - 1$ et $P_e(K_{v_i}[s^p]) < s_i^p$.

Exemple 7.2.3 – Prenons une paramétrisation P_{e^0} quelconque possible dans l'environnement e^0 . Les paramètres opérables identifiés sont : $K_{Operon} = 0$, $K_{Operon, \mathbf{free}} = 1$, $K_{Operon, \mathbf{alg}} = 1$, $K_{Operon, \mathbf{alg free}} = 2$, $K_{MucB} = 0$, $K_{MucB, \mathbf{prod}} = 1$, $K_{Calcium} = 1$, $K_{Calcium, \mathbf{A}} = 1$.

On en déduit le graphe de transitions asynchrone, *GTA* pour l'environnement e^0 et la paramétrisation P_{e^0} :

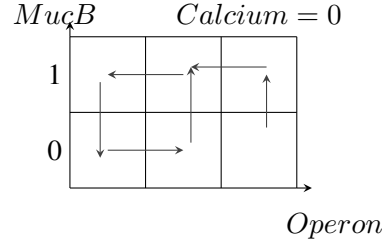


Figure 7.5 – GTA pour *Pseudomonas aeruginosa* dans l'environnement e^0 .

Par défaut si l'on prend en compte le graphe de régulation global, le nombre de paramétrisations pour *Operon* est de 6561 (nombre déjà calculé dans l'exemple 7.1.6). Regardons maintenant l'impact sur le nombre de paramétrisations lorsque l'on se place dans un environnement. Le nombre d'entrées de la variable *Operon* est réduit de 1. Le nombre de paramétrisations pour *Operon* devient donc $3^{(2^2)} = 81$. Le nombre de paramétrisations à énumérer est donc exponentiellement réduit tout comme la taille de graphe du transition en fonction du nombre de variables d'environnement.

7.2.2 Formules et abstractions de modèles

Il est maintenant crucial de bien définir chaque propriété environnementale (e, φ) . A partir du réseau de régulation dans un environnement \mathcal{N}_e et d'une paramétrisation $P \in \mathcal{N}_e$, on construit le graphe de transitions associé et le model-checking pour la formule φ est lancé.

7.2.2.1 Modèles d'un environnement

La procédure de décision pour savoir si φ est satisfaite dans l'environnement e reste identique au cas du réseau de régulation global. La seule nuance est que la procédure de décision sera associée à un environnement particulier et permettra d'obtenir des modèles dans un environnement particulier.

Notation – Étant donné un réseau de régulation dans un environnement \mathcal{N}_e , une paramétrisation pour ce réseau $P \in P_{\mathcal{N}_e}$, le langage $\mathcal{L}_{V \setminus VE}$ de formules *CTL* sur les variables internes $V \setminus VE$, le model checking $check_{\mathcal{N}_e}$ est définie par :

$$check_{\mathcal{N}_e} : P_{\mathcal{N}_e} \times \mathcal{L}_{V \setminus VE} \rightarrow \text{Bool}$$

tel que $check_{\mathcal{N}_e}(P, \varphi) = true$ ssi φ est vrai sur GTA_P

On définit maintenant les modèles d'une propriété environnementale.

Définition 7.2.5 (Modèles d'une propriété environnementale). Les modèles d'une propriété environnementale (e, φ) correspondent à l'ensemble des paramétrisations qui valident φ dans \mathcal{N}_e : $\mathcal{M}_e(\varphi) = \{P \in P_{\mathcal{N}_e} \mid check_{\mathcal{N}_e}(P, \varphi) = true\}$.

Ainsi on est capable d'obtenir les modèles environnement par environnement. Nous souhaitons maintenant calculer les modèles du réseau de régulation global.

7.2.2.2 Abstraction de paramétrisations et de modèles

Afin de pouvoir combiner les modèles $\mathcal{M}_{e^i}(\varphi^i)$ pour plusieurs e^i , qui se rapportent à différents jeux de paramètres, nous devons faire l'abstraction de l'ensemble des paramètres opérables à un sur-ensemble commun (l'ensemble des paramètres opérables du réseau global). Puisque $\mathcal{K}_{e^i} \subset \mathcal{K}$ pour tous les e^i , chaque paramétrisation P_{e^i} est abstraite par un sous-ensemble de paramètres dans $P_{\mathcal{N}}$.

Définition 7.2.6 (Abstraction d'une paramétrisation). Soit $P_{\mathcal{N}}$ l'ensemble des paramétrisations du réseau de régulation avec variables d'environnement $\mathcal{N} = (G, \mathcal{K})$. Soit $\mathcal{N}_e = (G, \mathcal{K}_e)$ le réseau de régulation pour l'environnement particulier $e \in E$, et $P_{\mathcal{N}_e}$ son ensemble de paramétrisations. L'abstraction d'une paramétrisation $P \in P_{\mathcal{N}_e}$ à $P_{\mathcal{N}}$ est le sous-ensemble $\mathcal{AP}_P \subset P_{\mathcal{N}}$ tel que : $\forall P' \in \mathcal{AP}_P, \forall K_{v,\omega} \in \mathcal{K}_e, P'(K_{v,\omega}) = P(K_{v,\omega})$, et $\forall K_{v,\omega} \notin \mathcal{K}_e, P'(K_{v,\omega}) \in \llbracket 0, b_v \rrbracket$.

En d'autres termes, P est la *projection* de \mathcal{AP} .

Définition 7.2.7 (Abstraction de modèles). L'abstraction de modèles d'une propriété environnementale (e, φ) est l'union des abstractions des paramétrisations dans $\mathcal{M}_e(\varphi)$:

$$\mathcal{AM}_e(\varphi) = \bigcup_{P \in \mathcal{M}_e(\varphi)} \mathcal{AP}_P.$$

Étant donnée une liste de propriétés environnementales $[(e^i, \varphi^i)]_{i=1}^n$, les paramètres satisfaisants toutes ces propriétés sont l'intersection des abstractions des modèles de chaque (e^i, φ^i) .

Définition 7.2.8 (Modèles de propriétés environnementales). Soit $\Psi = [(e^i, \varphi^i)]_{i=1}^n$ une liste de propriétés environnementales. Les modèles de Ψ correspondent à l'ensemble :

$$\mathcal{M}(\Psi) = \bigcap_{i=1}^n \mathcal{AM}_{e^i}(\varphi^i).$$

Exemple 7.2.4 – Soit $\mathcal{K} = \{K_1, K_2, K_3, K_4\}$ les paramètres de \mathcal{N} avec $D_1 = \llbracket 0, 1 \rrbracket$, $D_2 = \llbracket 0, 2 \rrbracket$, $D_3 = \llbracket 0, 3 \rrbracket$ et $D_4 = \llbracket 0, 1 \rrbracket$ les domaines de leurs variables associées. Supposons que $\mathcal{K}_{e^1} = \{K_1, K_4\}$ et $\mathcal{K}_{e^2} = \{K_1, K_2\}$ sont les paramètres opérables pour les environnements e^1 et e^2 .

- Soit $P_1 \in P_{\mathcal{N}_{e^1}}$ une paramétrisation de \mathcal{N}_{e^1} qui attribue 0 à K_1 et 1 à K_4 (notée $P_1 = (0, -, -, 1)$).
- Soit $P_2 \in P_{\mathcal{N}_{e^2}}$, une paramétrisation de \mathcal{N}_{e^2} qui attribue 0 à K_1 et 1 à K_2 (notée $P_2 = (0, 1, -, -)$).

Alors $AP = (0, 1, 3, 1) \in P_{\mathcal{N}}$ qui assigne 0 à K_1 , 1 à K_2 , 3 à K_3 et 1 à K_4 appartient à \mathcal{AP}_1 et à \mathcal{AP}_2 . De plus, si P_1 est un modèle de (e^1, φ^1) et P_2 est un modèle de (e^2, φ^2) , alors AP

est un modèle cohérent avec l'ensemble de la liste des propriétés environnementales $[(e^i, \varphi^i)]_{i=1}^2$. $AP' = (0, 0, 2, 1)$ appartient aussi à \mathcal{AP}_{P_1} mais pas à \mathcal{AP}_{P_2} . Dans ce cas même si P_1 est un modèle de (e^1, φ^1) , la paramétrisation AP' ne sera pas valide pour obtenir un modèle pour toutes les propriétés environnementales car elle ne fait pas partie de \mathcal{AP}_{P_2} .

7.2.3 Équivalence des approches

La question qui se pose maintenant est de savoir si les deux méthodes sont équivalentes et valident donc les mêmes modèles (mêmes paramétrisations). La réponse repose d'une part sur le lien entre le graphe de transitions obtenu pour un environnement et le sous-graphe du GTA global induit par un environnement, et d'autre part sur le lien entre la formule globale et les formules environnementales.

Nous démontrerons dans un premier temps l'isomorphisme des graphes de transitions entre celui obtenu pour un environnement donné et le sous-graphe issu du graphe global dans le même environnement.

7.2.3.1 Isomorphisme des graphes d'états

Lemme 7.2.1 (Isomorphisme entre GTA_P et un sous-graphe de $GTA_{P'}$). *Pour chaque environnement e , chaque paramétrage P de \mathcal{PN}_e et chaque $P' \in \mathcal{AP}_P$, il existe un isomorphisme canonique entre GTA_P et le sous-graphe de $GTA_{P'} \in \mathcal{N}_{global}$ réduit à S_e .*

Preuve du lemme.

Chaque état du sous-graphe de $GTA_{P'}$ réduit à S_e est trivialement associé à un état de GTA_P (voir Def. 3.2.1 et 7.2.1). Montrons que les transitions sont les mêmes. Considérons un état commun s .

Pour déterminer les paramètres applicables de s dans GTA_P , on doit évaluer les formules des multiplexes contrôlant chaque variable interne (qui n'est pas environnementale). Les atomes sont évalués de la même manière dans GTA_P et $GTA_{P'}$ (le n-uplet représentant s dans GTA_P est égal à celui représentant s dans $GTA_{P'}$). Ainsi, les paramètres applicables des variables internes dans s sont les mêmes, conduisant aux mêmes transitions qui ne changent pas les variables d'environnement.

De plus, en raison de notre choix de valeurs de paramètres pour contrôler l'évolution des variables d'environnement, il n'existe aucune transition dans $GTA_{P'}$ qui modifie les valeurs des variables d'environnement.

Ainsi, lorsque toutes les variables d'environnement sont fixées, le sous-graphe de $GTA_{P'}$ réduit aux états correspondant à l'environnement e , et GTA_P sont isomorphes.

□

Il ne reste donc plus qu'à prouver l'équivalence des modèles.

7.2.3.2 Équivalence des modèles

Le théorème suivant affirme que les deux approches conduisent au même ensemble de modèles de propriétés environnementales.

Théorème 7.2.2 ($\mathcal{M}(\Psi) = \mathcal{M}(\Phi_{global})$). *Étant donné une liste $\Psi \equiv [(\varphi^i, e^i)]_{i=1}^n$ de propriétés environnementales sur un graphe de régulation avec des variables d'environnement $G = (V, VE, M, A)$, l'ensemble $\mathcal{M}(\Psi)$ de modèles de Ψ (calculés environnement par environnement) est égal à l'ensemble des modèles de Φ_{global} sur \mathcal{N}_{global} .*

Démonstration.

- (1) Considérons un ensemble de paramétrisations P sélectionnées par l'approche environnement par environnement. $P \in \bigcap_{i=1}^n \mathcal{AM}_{e^i}(\varphi^i)$ (voir Def. 7.2.8). Pour tout $i \in \llbracket 1, n \rrbracket$, φ^i est satisfaite dans tous les états de GTA_{P_i} où P_i est la projection de P sur les paramètres de l'environnement e^i . Alors, par Lemme 7.2.1, φ^i est satisfaite dans les états correspondant à e^i dans GTA_P . La formule $\varepsilon^i \Rightarrow \varphi^i$ est alors satisfaite dans tous les états de GTA_P . Puisque $\Phi_{global} \equiv \bigwedge_{i=1}^n (\varepsilon^i \Rightarrow \varphi^i)$, nous concluons que Φ_{global} est satisfait ($\mathcal{M}(\Psi) \subset \mathcal{M}(\Phi_{global})$).
- (2) Inversement, considérons maintenant un ensemble de paramètres P sélectionnés par l'approche classique : $P \in \mathcal{M}(\Phi_{global})$. Puisque $\Phi_{global} \equiv \bigwedge_{i=1}^n (\varepsilon^i \Rightarrow \varphi^i)$, pour tous les $i \in \llbracket 1, n \rrbracket$, φ^i est satisfait dans tous les états définissant l'environnement e^i . Ainsi, par le Lemme 7.2.1, $P \in \bigcap_{i=1}^n \mathcal{AM}_{e^i}(\varphi^i)$, en d'autres termes, $\mathcal{M}(\Phi_{global}) \subset \mathcal{M}(\Psi)$.

□

7.2.4 Validation environnement par environnement via *TotemBioNet*

7.2.4.1 Validation des modèles des environnements

L'approche environnement par environnement a aussi été implémentée dans *TotemBioNet*. La notion de variables d'environnements a été ajoutée récemment au langage d'entrée de *TotemBioNet*. Un fichier d'entrée de *TotemBioNet* peut alors décrire un graphe de régulation avec des variables d'environnements, utilisables pour plusieurs environnements. Pour un environnement spécifique e , les valeurs des variables d'environnement doivent être définies à leurs valeurs spécifiques afin de calculer les modèles $\mathcal{M}_e(\varphi)$ pour une propriété φ . Une étape de pré-traitement construit l'ensemble des paramètres *opérables* dans \mathcal{N}_e elle permet d'ignorer tous les autres paramètres. Ceci réduit exponentiellement le nombre de paramètres qui doivent être énumérés pour calculer $\mathcal{M}_e(\varphi)$.

Partant d'un graphe de régulation avec des variables d'environnement et une liste de propriétés environnementales $\Psi = [(e^i, \varphi^i)]_{i=1}^n$, *TotemBioNet* doit être appelé successivement n fois pour chaque couple (e^i, φ^i) . Chaque appel à *TotemBioNet* génère un fichier « .csv » qui contient les paramétrisations pour que φ^i soit vérifiée dans l'environnement e^i . En d'autres termes il contient $\mathcal{M}_{e^i}(\varphi^i)$. Ces fichiers « .csv » doivent être ensuite combinés afin de calculer les modèles de Ψ .

7.2.4.2 Combinaisons des modèles environnementaux

Comme expliqué précédemment (voir Def. 7.2.2), les paramètres opérables ne sont pas les mêmes pour chaque e^i . Ainsi, une simple intersection des fichiers « .csv » qui ont été obtenus indépendamment pour chaque e^i n'est pas suffisante. Il est d'abord nécessaire d'abstraire chaque ensemble spécifique de paramètres opérables à l'ensemble plus général de tous les paramètres du graphe de régulations (voir Def 7.2.5 et 7.2.8). L'implémentation des opérations permettant la combinaison de modèles a été effectuée par Hélène Collavizza et Cyril Guillot, étudiant en sciences de l'ingénieur à Polytech Nice Sophia-Antipolis. Elle utilise les *diagrammes de décision multivalués* (MDD).

Les MDDs fournissent une représentation compacte d'un ensemble des données discrètes, ainsi que des opérations efficaces sur les ensembles : intersection, union et complémentation. Il est facile (et efficace) d'exprimer qu'à l'intérieur d'un ensemble commun de variables, certaines peuvent prendre n'importe quelle valeur. Ainsi, il est très pratique d'abstraire et de combiner des ensembles de paramétrisations qui ne concernent pas les mêmes paramètres. Plus précisément, un MDD représentant $\mathcal{AM}_{e^i}(\varphi_i)$ est construit pour chaque environnement e^i . Les variables de ces MDDs sont les paramètres du système global. Les paramètres non opérables pour e^i peuvent prendre n'importe quelle valeur, tandis que les valeurs des paramètres opérables sont lues à partir du fichier « .csv » des modèles $\mathcal{M}_{e^i}(\varphi_i)$. L'intersection de tous les MDDs donne les modèles de Ψ . *TotemBioNet* fait appel à la bibliothèque *Colomoto mddlib*, bibliothèque² développée par A. Naldi, pour calculer l'intersection des modèles obtenus dans des environnements successifs. Cette bibliothèque a été conçue pour la modélisation de systèmes biologiques, et notamment pour trouver des états stables et analyser des circuits [Naldi et al., 2007].

Exemple 7.2.5 – Illustrons maintenant cette approche sur *Pseudomonas aeruginosa*. Parce que la variable d'environnement *Calcium* ne peut prendre que deux valeurs, seuls deux réseaux de régulation environnementaux sont construits. La formule φ^0 doit être vérifiée sur le premier, alors que φ^1 sur le second.

1. Dans l'environnement e^0 :
 - Le nombre de paramétrisations pour *Operon* est de 81 (cf. exemple 7.2.3).
 - La variable *MucB* n'est pas directement régulée par une variable d'environnement, donc le nombre de paramétrisations ne change pas par rapport au réseau de régulation global il est donc de 4 (cf. 7.1.6).
 Ainsi, pour l'environnement e^0 , la formule φ^0 doit être vérifiée sur $81 \times 4 = 324$ paramétrisations.
2. Pour l'environnement e^1 , l'énumération des paramétrisations donne le même nombre : il y a 324 paramétrisations pour lesquels une vérification de φ^1 doit être effectuée.

Il y a donc en tout $324 + 324 = 648$ paramétrisations à considérer (à comparer aux 26 244 paramètres à considérer dans l'approche globale). Chacune de ces paramétrisations est confrontée à la propriété d'intérêt dans l'environnement considéré par *TotemBioNet* en environ 700 millisecondes. L'intersection des ensembles de modèles (en utilisant les MDDs) nécessite 9,61 millise-

2. <https://github.com/colomoto/mddlib>

condes. Le temps total pour calculer les modèles compatibles pour les deux environnements est de 1,41 secondes³, ce qui est 105 fois plus rapide que l’approche globale présentée en la Sec. 7.1.4.

Cet exemple est très petit, mais il montre déjà que la seconde approche (environnement par environnement) est moins gourmande en temps. Avec un réseau de régulation plus important impliquant de nombreuses variables d’environnement, la pertinence de la seconde approche est encore plus visible. Néanmoins, en pratique son utilisation nécessite la répétition de n lancements de *TotemBioNet* (n correspondant au nombre d’environnements). Lorsque ce nombre d’environnement est grand, une automatisation de la chaîne de traitement est nécessaire : c’est l’objet de l’outil développé *GreenBioNet*.

7.3 Automatisation de la méthode environnement par environnement

L’automatisation des différentes étapes de la méthode par environnement nécessite la création au préalable de tous les fichiers nécessaires pour la validation des modèles dans chaque environnement. Il faut associer les comportements (φ^i) à ces différents fichiers. Pour cela nous avons à disposition la matrice de validation.

7.3.1 Utilisation de la matrice de validation

Une matrice de validation, dont la définition 5.4.1 est donnée au chapitre 5, met à disposition la liste des environnements (valeurs pour chacune des variables d’environnement), ainsi que les comportements dans ces différents environnements. Dans une matrice de validation, chaque ligne représente un environnement et chaque colonne un marqueur biologique associé à une variable. À l’aide du graphe de régulation global et des paramètres partiellement identifiés, on va alors pouvoir calculer les paramétrisations menant à des graphes de transitions compatibles avec une matrice de validation donnée.

En se référant aux définitions apportées 5.2.1, 5.4.2 page 51 et 55 du chapitre 5, nous pouvons définir la notion de modèle pour une matrice de validation.

Définition 7.3.1 (Modèle d’une matrice de validation). *Le modèle d’une matrice de validation MP est l’ensemble :* $\mathcal{M}([\Phi_e^{MP}, e]_{e \in E}) = \bigcap_{e \in E} \mathcal{AM}_e(\Phi_e^{MP})$, où Φ_e^{MP} est la propriété environnementale de la matrice MP reprenant toutes les propriétés connues dans l’environnement e et E l’ensemble des environnements considéré.

7.3.2 Automatisation par *GreenBioNet*

L’outil *GreenBioNet* (disponible sur le dépôt GIT à l’adresse <https://gitlab.com/totembionet/greenbionet>) prend en entrée : le graphe de régulation GR , une matrice de validation associée et l’ensemble des paramètres (partiellement ou totalement) identifiés. La sortie de *GreenBioNet* consiste en l’ensemble des modèles du réseau global qui satisfont toutes les

3. Les temps d’exécution sont calculés de la même manière que pour l’approche globale et les lecteurs intéressés peuvent obtenir les fichiers d’entrée et les résultats dans le dépôt : <https://gitlab.com/totembionet/totembionet/-/tree/master/examples/CMSB2021>

propriétés environnementales. Comme dit précédemment, l’outil est basé sur des appels successifs de *TotemBioNet*. La syntaxe des fichiers d’entrée et de sortie, à l’exception du fichier d’entrée décrivant la matrice de validation (qui a un traitement spécifique), est cohérente avec la syntaxe des fichiers d’entrée de *TotemBioNet*⁴. La figure 7.6 présente le workflow de *GreenBioNet*.

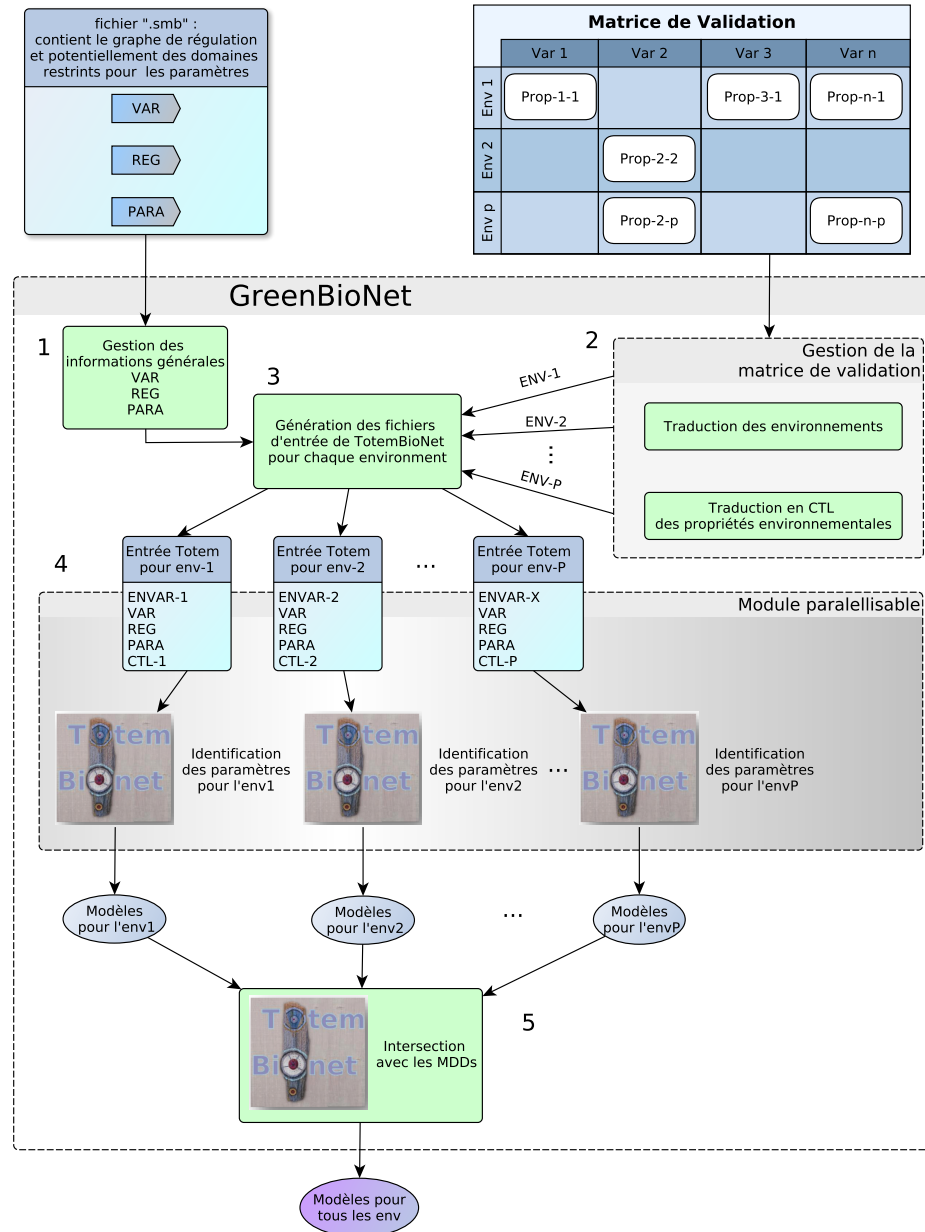


Figure 7.6 – *GreenBioNet* Workflow : automatisation de l’approche « diviser pour régner » pour obtenir le modèle d’une matrice de validation.

4. voir le manuel d’utilisation de *TotemBioNet* : <https://gitlab.com/totembionet/totembionet/-/blob/master/doc/userManual.pdf>

Le workflow de *GreenBioNet* est divisé en cinq parties décrites ci-dessous.

1. Traiter les informations concernant le graphe de régulation et les domaines (potentiellement réduits) des différents paramètres du réseau de régulation global,
2. Extraire de la matrice de validation chaque propriété environnementale,
3. Créer pour chaque environnement un fichier d'entrée *TotemBioNet* contenant le réseau de régulation dédié à cet environnement, les domaines de paramètres potentiellement réduits et la propriété environnementale associée,
4. Lancer *TotemBioNet* successivement pour chaque fichier d'entrée créé, afin de calculer les modèles pour cet environnement,
5. Lancer une dernière fois *TotemBioNet* pour calculer l'intersection des abstractions des modèles précédemment obtenus.

7.3.2.1 Traitement de l'information indépendante des environnements [Fig.7.6 (1)]

Tout d'abord, l'outil traite les informations communes à tous les environnements qui sont indiquées dans les différents blocs du fichier d'entrée (avec l'extension `.smb`), cf. Figure 7.7-haut. Le premier bloc liste toutes les variables avec leurs domaines (bloc `VAR`) ; le bloc `REG` décrit les régulations ; et le bloc `PARA` spécifie les domaines potentiellement réduits des paramètres (lorsque des études précédentes ont réduit les domaines initiaux). Toutes ces informations apparaîtront ensuite dans chacun des fichiers d'entrée *TotemBioNet* créés.

7.3.2.2 Extraction des propriétés de la matrice de validation [Fig.7.6 (2)]

La matrice de validation est traitée ligne par ligne, c'est-à-dire environnement par environnement. Chaque ligne correspond à un fichier d'entrée *TotemBioNet* particulier. La traduction de chaque ligne de la matrice de validation se fait en trois étapes :

- La première concerne les variables d'environnement : leurs valeurs fixes sont enregistrées et sont traduites sous la forme d'un bloc "ENVAR". dans chaque fichier d'entrée *TotemBioNet*. Ce bloc spécifie les valeurs des variables d'environnement.
- La deuxième étape consiste à traduire chaque case de la ligne considérée en *Fair-Path CTL*. Lors de l'analyse d'une case de la matrice de validation, chaque motif, comme « OSC », « td(.) », est automatiquement remplacé par la formule *Fair-Path CTL* équivalente en prenant soin de l'adapter à la variable concernée. Par exemple, dans la ligne 2 de la matrice de validation donné en bas de la figure 7.7, le motif « td(2) » dans la colonne **Operon**, sera traduit en la formule *Fair-Path CTL* : $A'F(A'G(\text{Operon} = 2))$. S'il y a une négation, « ! », la formule *Fair-Path CTL* précédente est modifiée en conséquence.
- Enfin, une fois que tous les éléments de la ligne sont traités, la conjonction des formules obtenues est construite et correspond à la propriété associée à l'environnement.

Notons que si une case de la matrice n'est pas remplie alors l'outil n'interprète aucune formule correspondante, ce qui revient à interpréter cette case comme la formule « Vrai ».

7.3.2.3 Création des fichiers d'entrée *TotemBioNet* [Fig.7.6 (3)]

La troisième étape consiste à prendre toutes les informations traduites provenant à la fois de la matrice de validation et du fichier `.smb` et à créer les fichiers d'entrée *TotemBioNet*. Les

différents fichiers d'entrée créés fournissent le même *GR*, dans lequel les variables d'environnement n'apparaissent que dans les formules des régulations, et les paramètres partiellement identifiés (les blocs VAR, REG et PARA ne dépendent pas de l'environnement). Les environnements traduits (bloc ENVAR) et leurs formules associées (bloc FAIRCTL) provenant de la matrice de validation, permettent la spécification complète de chaque fichier d'entrée pour le lancement de *TotemBioNet*.

7.3.2.4 Lancements successifs de *TotemBioNet* [Fig.7.6 (4)]

TotemBioNet est lancé pour chaque fichier obtenu à l'étape précédente. Pour chaque fichier, *TotemBioNet* renvoie un fichier « .csv » dans lequel des colonnes représentent les différents paramètres opérables et chaque ligne une paramétrisation menant à un *GTA* présentant la propriété environnementale spécifiée. Cette étape contraignante (en raison du grand nombre possible de paramétrisations à énumérer) peut être améliorée. En effet, on peut envisager une parallélisation des lancements de *TotemBioNet* car les appels au model-checking sont indépendants.

7.3.2.5 Intersection des abstractions de modèles [Fig.7.6 (5)]

La dernière étape consiste à calculer toutes les paramétrisations qui satisfont tous les comportements dépendant de l'environnement, décrits dans la matrice de validation. Cette opération est également effectuée par *TotemBioNet* en utilisant son option « intersection » : elle permet l'abstraction et l'intersection de tous les résultats obtenus par *TotemBioNet* environnement par environnement. Enfin, *GreenBioNet* renvoie en sortie un fichier « .csv » qui liste de manière exhaustive tous les modèles compatibles avec toutes les lignes de la matrice de validation.

7.3.3 Exemple

La figure 7.7 fournit une partie du fichier d'entrée « .smb » représentant la description structurale du réseau de régulation ainsi qu'une partie de la matrice de validation, qui est simplement un fichier « .csv ».

7.4 Conclusion

Au cours de ce chapitre, deux méthodes de validation ont été exposées. La première, la méthode globale, permet d'énumérer les paramétrisations du réseau de régulation global et de ne retenir que les paramétrisations qui mènent à un graphe de transitions satisfaisant toutes les propriétés décrites dans la matrice de validation.

Néanmoins nous proposons une seconde méthode qui extrait successivement les modèles dans différents environnements pour ensuite combiner les résultats et obtenir l'ensemble des paramétrisations cohérentes pour tous les environnements. Cette seconde méthode est plus efficace que la première car elle réduit le nombre de paramétrisations à explorer. Comme le temps d'exécution global dépend quasi-linéairement du nombre de paramétrisations associé au réseau, le temps d'exécution global est fortement réduit. Cela nous permet de traiter des exemples de taille raisonnable qui n'étaient pas encore accessibles à la méthode globale.

```

VAR
Operon = 0..2 ;
MucB = 0..1 ;
REG
prod [(Operon>=1)] => mucuB;
free [!(mucuB>=1)] => Operon;
alg [Operon>=2] => Operon;
ca [calcium>=1] => Operon;
PARA
K_Operon = 0 ;
K_Operon:free = 1 ;

```

(haut) Une partie du fichier « .smb » donné comme entrée à *GreenBioNet*. Le fichier est organisé en trois blocs : le bloc VAR énumère les variables avec leurs domaines, le bloc REG décrit les régulations à travers la notion de multiplexe, et le bloc PARA spécifie les domaines potentiellement réduits des paramètres. Le fichier d'entrée est écrit en accord avec la syntaxe de *TotemBioNet*.

Contexte	Phénotype	
ENV	VAR	
Calcium	Operon	MucB
0	$((Operon = 2) \Rightarrow A'G((Operon = 2))) \&$ $((Operon < 2) \Rightarrow A'G((Operon < 2)))$	
1	td(2)	

(Bas) Matrice de validation donnée en entrée à *GreenBioNet*.

Figure 7.7 – Fichiers d'entrée donnés à *TotemBioNet* pour la validation du modèle de *Pseudomonas aeruginosa*.

Cependant cette méthode de validation environnement à certaines limites. Notamment, lorsque les seuls paramètres fixés sont ceux des variables sous le contrôle des variables d'environnement, cette seconde méthode va lancer un appel à *TotemBioNet* pour chaque environnement, mais le nombre de paramétrisations pour chaque environnement est le même que pour la méthode globale.

Même si la phase d'énumération des paramétrisations à été améliorée, elle reste limitante. On pourrait alors envisager une parallélisation à gros grain afin de diminuer le temps de calcul. En effet, tous les appels au model checker sont indépendants, ils pourraient donc être fait en parallèle.

Cette méthode reste néanmoins avantageuse car elle permet, lorsque aucune paramétrisation n'est cohérente avec les spécifications données, d'avoir un retour sur expérience plus précis. En effet, comme on définit des modèles pour chaque environnement, on peut retrouver quel environnement n'est pas compatible avec les spécifications. On peut alors retravailler sur le contexte de modélisation pour cet environnement.

Modélisation par enrichissements successifs

La construction d'un modèle de grande envergure est aujourd'hui un travail complexe car malgré la présence de méthodes de vérification automatique, le nombre de paramétrisations est tel que l'on n'est pas capable d'en énumérer la totalité en un temps convenable. Même si les systèmes complexes ne sont pas incrémentaux par nature, cette limitation peut être contournée par l'enrichissement de modèles préexistants. Ce chapitre décrit différents types d'enrichissements possibles et la méthode à suivre pour effectuer chacun d'entre eux dans le cadre de la théorie de R.Thomas.

8.1	Insérer un multiplexe	86
8.1.1	Motivation biologique d'un tel enrichissement	87
8.1.2	Modification du graphe de régulation	88
8.1.3	Modification cohérente des paramètres	88
8.1.4	Non régression et révision cohérente de la matrice de validation	90
8.2	Insérer une variable	91
8.2.1	Motivation biologique d'un tel enrichissement	91
8.2.2	Modification du graphe de régulation	92
8.2.3	Modification cohérente des paramètres	92
8.2.4	Non régression et révision cohérente de la matrice de validation	93
8.3	Ajouter un niveau à une variable préexistante	94
8.3.1	Motivation biologique d'un tel enrichissement	94
8.3.2	Modification du graphe de régulation	95
8.3.3	Modification cohérente des paramètres	96
8.3.4	Non régression et révision cohérente de la matrice de validation	97
8.4	Scinder une variable ou un multiplexe préexistant	98
8.4.1	Motivation biologique d'un tel enrichissement	98
8.4.2	Modification du graphe de régulation	99
8.4.3	Modification cohérente des paramètres	102
8.4.4	Non régression et révision cohérente de la matrice de validation	104

L'arrivée d'outils d'aide à la conception de modèles par des méthodes formelles a propulsé la modélisation vers des systèmes de plus en plus gros. Aujourd'hui on en arrive à modéliser des systèmes de taille tellement importante que même *via* l'utilisation de *TotemBioNet*, le nombre de paramétrisations potentielles, est trop grand, si l'on n'estime pas une grande partie des paramètres à la main. Le développement de modèles *de novo* *via* l'approche du cycle en « V » présentée dans le chapitre 4 de la partie II du manuscrit n'est plus suffisante aujourd'hui, et en particulier dans le cadre de la représentation des régulations du métabolisme cellulaire dans le cancer du pancréas. On se retrouve donc face à un nouveau problème que l'on peut résoudre au moins en partie en ayant une approche durable : la réutilisation d'un modèle préexistant que l'on va enrichir. Cette approche basée sur le recyclage dont la méthodologie est décrite section 4.2.1 chapitre 4 permet notamment la spécialisation d'un modèle à une problématique particulière. Pour ce faire, nous avons suivi un nouveau cycle en spirale qui vise cette fois-ci à faire des incréments de petites taille, qui pourront être facilement réévalués s'ils ne permettent pas d'obtenir des modèles cohérents. Dans ce chapitre nous décrivons les primitives de l'enrichissement incrémental : comment insérer un multiplexe, une variable mais aussi comment ajouter un niveau ou scinder une variable. On peut imaginer d'autres enrichissements, même si le descriptif paraît aujourd'hui relativement complet.

Quel que soit l'enrichissement que l'on va faire, on va suivre une méthodologie bien définie :

1. On identifie ce que l'on doit ajouter qui n'est pas dans le modèle (*i.e.* contexte de modélisation) avec les éléments associés (on explique le concept biologiquement),
2. On modifie le graphe en conséquence,
3. On modifie, en cohérence avec le nouveau graphe, les paramètres impliqués dans l'enrichissement,
4. On réévalue la matrice de validation par l'ajout ou la modification de comportements potentiellement connus.
5. Puis l'on valide l'enrichissement *via* *TotemBioNet*.

L'ensemble du chapitre est organisé selon ce plan pour chaque type d'enrichissement décrit. Notons que la dernière étape n'est pas explicitée dans le plan car nous ne décrivons pas dans ce chapitre des cas pratiques mais des étapes méthodologiques pour enrichir un modèle existant. Nous utiliserons, à titre d'illustration, le modèle de *Pseudomonas aeruginosa* présenté dans le chapitre 3 dont le réseau de régulation et la matrice de validation sont rappelés Figure 8.1, et table 8.1 ci-dessous.

Tous les motifs ou formules logiques indiqués dans les matrices qui figurent dans ce chapitre représentent des comportements biologiques formalisés à l'aide de logique temporelle *Fair-Path CTL*. L'explication et la traduction de ces motifs est disponible dans le chapitre 5. Le pattern « td(2) » pour la variable *Operon* est traduit par la formule *Fair-Path CTL* : $AF(AG(Operon \geq 2))$ (voir section 5.3.2).

8.1 Insérer un multiplexe

L'insertion d'un multiplexe ou la modification de la formule d'un multiplexe déjà existant est un des fondamentaux de l'enrichissement de modèles dans le formalisme de René Thomas. En

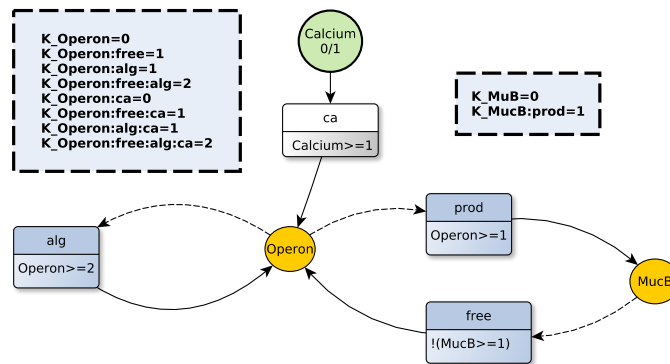


Figure 8.1 – Réseau de régulation de *Pseudomonas aeruginosa* présenté chapitre 3. Les paramètres sont indiqués dans les rectangles en pointillés en syntaxe concrète de *TotemBioNet*

Environnement	Phénotypes	
Calcium	Operon	MucB
0	$((Operon = 2) \Rightarrow A'G((Operon = 2)))$ $((Operon < 2) \Rightarrow A'G((Operon < 2)))$	
1	td(2)	

Table 8.1 – Matrice de validation pour le modèle de *Pseudomonas aeruginosa*.

effet, les multiplexes sont très souvent réévalués puisque la majorité des enrichissements présentés dans ce chapitre entraînent aussi la modification/création de ces derniers.

8.1.1 Motivation biologique d'un tel enrichissement

L'ajout d'un multiplexe est motivé par l'absence d'une régulation dans le réseau de régulation vis à vis du système représenté. Pour pouvoir faire l'enrichissement, il faut définir le/les éléments régulés dans le modèle et le type de régulation (coopération par exemple). Après avoir défini l'ensemble des éléments impliqués on va pouvoir formaliser ces nouveautés dans le graphe de régulation.

Outre la modélisation d'une nouvelle connaissance sur une régulation, nous pouvons, à l'aide d'un multiplexe, modéliser une connaissance qui est liée aux transitions entre les états du système. Par exemple, supposons que nous souhaitons interdire une transition qui est biologiquement incorrecte. Nous pouvons modifier la formule d'un multiplexe qui joue sur la variable évoluant vers l'état non souhaité. La modification de la formule du multiplexe impacte l'ensemble des états où la formule est vérifiée. Dès lors d'après le formalisme utilisé, lorsque dans un état le multiplexe est vérifié, il devient ressource (cf chapitre 3 section 3.2). Cela a pour conséquence la modification des paramètres applicables dans la constitution du graphe de transition qui est à son tour modifié. Finalement, quand on sait qu'une transition dans le graphe de transition doit être interdite (ou autorisée), on peut donc modifier ou concevoir une formule d'un multiplexe pour obtenir la dynamique correcte.

8.1.2 Modification du graphe de régulation

Après avoir défini biologiquement ce que l'on veut représenter, il faut à partir des connaissances formaliser le contenu du multiplexe. Deux choses sont à définir : le nom, qui doit indiquer au mieux la régulation représentée, et la formule qui indique quelles sont les conditions pour que la régulation s'effectue. Pour la conception de la formule logique on utilise les connaissances biologiques disponibles sur la régulation. Pour cela on identifiera notamment les entités qui coopèrent et les variables/multiplexes par lesquelles ces entités sont représentées dans le modèle à enrichir. Une fois construit, nous devons indiquer quelles sont les cibles du multiplexe inséré. Une fois l'ensemble de ces informations défini, il suffit de modifier le graphe en conséquence : ajouter le multiplexe et sa formule, et le relier à ses régulateurs par un arc plein vers se(s) variable(s) cible(s).

Exemple 8.1.1 – Prenons l'exemple de *Pseudomonas aeruginosa* où l'on souhaite ajouter une régulation (figure 8.1). Nous supposons maintenant que la présence de calcium provoque l'activation de la protéine *MucB*. On va pouvoir ajouter un multiplexe nommé **hypo** (pour hypothèse) dont la formule est : $\text{Calcium} \geq 1$. On relie donc ce multiplexe à *MucB* par un arc plein (et bien sûr on ajoute un arc pointillé de calcium vers le multiplexe). Le graphe de régulation résultant de l'enrichissement du graphe de régulation de la figure 8.1 est donné en figure 8.2. Notons ici que dans un multiplexe, une action inhibitrice se note par une négation au début devant la formule associée (voir multiplexe **free**).

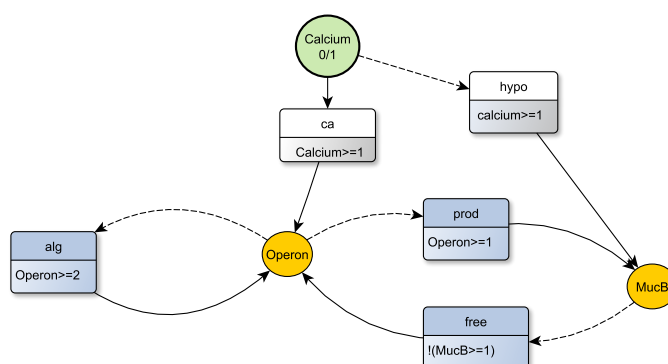


Figure 8.2 – Graphe de régulation de *Pseudomonas aeruginosa* avec ajout d'un multiplexe régulateur positif **hypo**.

8.1.3 Modification cohérente des paramètres

Lors de l'ajout d'un multiplexe, comme nous changeons les propriétés de régulation de certaines variables, il faut s'intéresser aux paramètres présents avant l'enrichissement et les réévaluer partiellement ou totalement selon certains critères que l'on définira pour conserver une cohérence dans l'abstraction des différents éléments biologiques par rapport au modèle préexistant. Pour savoir quel paramètre réévaluer, il faut savoir si avant l'ajout de la régulation, on se trouvait dans le cas où la régulation était toujours effective ou dans le cas où elle ne l'était pas.

Exemple 8.1.2 – Revenons à l’enrichissement précédent. L’ajout du multiplexe **hypo** implique que le nombre de paramètres pour **MucB** est doublé (car on ajoute un prédécesseur). Les paramètres du modèle enrichi qu’il faut déterminer sont donc : K_{MucB} , $K_{MucB,prod}$, $K_{MucB,hypo}$ et $K_{MucB,prodhypo}$. Certaines informations du modèle initial facilitent cette identification. Lors de la construction du modèle de *Pseudomonas aeruginosa* initial, l’estimation des paramètres s’est faite lorsque le calcium n’était pas impliqué : la ressource **prod**, et après enrichissement à le même sens abstrait. Autrement-dit, les paramètres K_{MucB} et $K_{MucB,prod}$ du modèle initial et leurs valeurs peuvent être reportés car leurs identifications ont été faites dans les mêmes conditions d’abstraction qu’avant l’enrichissement. Dans ce cas il ne reste plus qu’à identifier les deux derniers paramètres restants.

A l’inverse, imaginez que le réseau de régulation présupposait que le calcium était présent. Les paramètres alors estimés précédemment ne correspondent pas à la représentation des ressources indiquées. En effet, comme on a explicité le rôle du calcium, ce sont les paramètres ayant pour ressource **hypo**, soit $K_{MucB,hypo}$ et $K_{MucB,prodhypo}$, qui pourront potentiellement prendre les précédentes valeurs : $K_{MucB,hypo}$ prendra l’ancienne valeur de K_{MucB} et $K_{MucB,prodhypo}$ celle de $K_{MucB,prod}$. Les paramètres qu’il restera à identifier seront alors ceux présents dans le modèle avant enrichissement, soit (K_{MucB} et $K_{MucB,prod}$).

Nous n’avons traité de la modification des paramètres que dans le cas où la régulation ajoutée est positive. Sachant que la présence d’un inhibiteur revient à l’absence d’un activateur, lorsque l’on ajoute un multiplexe inhibiteur les règles que nous venons d’énoncer sont inversées.

Exemple 8.1.3 – Changeons la formule de l’exemple précédent pour définir un multiplexe inhibiteur représenté en figure 8.3.

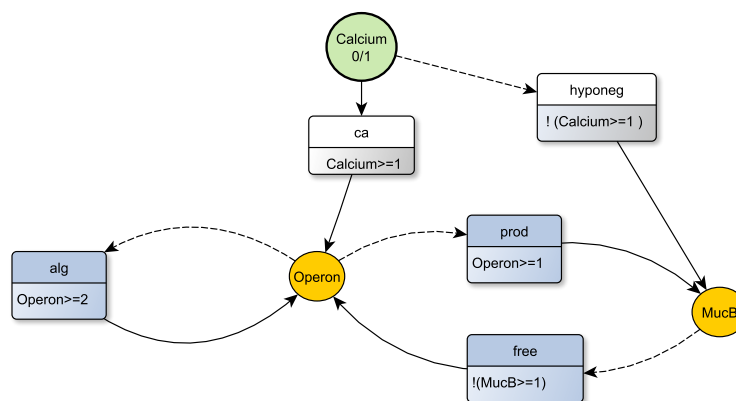


Figure 8.3 – Graphe de régulation après ajout d’un multiplexe inhibiteur **hyponeg**.

Dans ce cas, si l’on a estimé les paramètres en l’absence de calcium, alors le multiplexe est **hyponeg**. Cela sous-entend que la ressource était considérée comme toujours « présente » dans le modèle préexistant lors de l’estimation des paramètres et les paramètres initiaux (K_{MucB} et $K_{MucB,prod}$) doivent être ré-identifiés et les paramètres $K_{MucB,hypo}$ et $K_{MucB,prodhypo}$ pourront prendre les précédentes valeurs comme dans le cas où la régulation est positive. Toujours dans le cas où le multiplexe ajouté est une inhibition, si l’on avait préalablement estimé l’ensemble des

paramètres en présence de calcium alors les valeurs de paramètres initiaux peuvent être conservées.

Enfin, si la régulation est plus complexe et les conditions initiales d'estimation des valeurs de paramètres floues (on n'est pas sûr d'être toujours dans un cas où l'on considère le multiplexe vrai ou faux dans tous les états), il est plus judicieux d'utiliser des expériences de pensées pour réévaluer l'ensemble des paramètres. De plus, pour l'ensemble des paramètres il est nécessaire de vérifier la cohérence avec les conditions de Snoussi (vérifier qu'un paramètre avec plus de ressources a bien une valeur supérieure ou égale au paramètre avec moins de ressources soit, voir section 3.2.3).

La modification de la formule d'un multiplexe remet en cause les valeurs des paramètres car le sens biologique de la régulation peut être modifié. Au final, on conservera les valeurs des paramètres lorsque deux paramètres rendront compte de la même situation avant et après enrichissement (leur sens abstrait étant équivalent). Pour la suite lorsque qu'on rencontrera un cas similaire, on indiquera que les paramètres X avant enrichissement et Y après enrichissement sont *équivalents*.

8.1.4 Non régression et révision cohérente de la matrice de validation

La modification ou l'ajout d'un multiplexe ne joue que sur les transitions possibles et non sur les variables (ni leur nombre, ni leurs niveaux) : le comportement des variables n'a donc pas à être réévalué. Par conséquent, la matrice de validation associée au modèle préexistant n'a pas à être modifiée. Cependant, la modification de certains paramètres peut remettre en cause l'ensemble des comportements observés (un réseau de régulation est non modulaire). Il faut tout de même vérifier la non régression des propriétés définies pour le modèle avant enrichissement. Pour cela nous utilisons *TotemBioNet* pour confronter le réseau enrichi à la matrice de validation du modèle non enrichie (assimilable à une matrice de non régression).

Cependant, parfois l'ajout ou la modification d'un multiplexe est motivé par une connaissance biologique *phénotypique* complémentaire. Dans ce cas on peut compléter la matrice avec ces nouvelles connaissances pour gagner en précision sur le modèle.

Enfin, il existe un cas particulier où l'ajout d'un multiplexe modifie la matrice de manière drastique : si le multiplexe ajouté régule une variable d'environnement. Dans ce cas, la variable qui permettait de définir les environnements devient une variable interne du modèle car elle va varier en fonction de l'état du modèle. Formellement, la place de la variable dans la matrice est modifiée (voir chapitre 5 section 5.4). On peut alors définir des comportements pour la variable en fonction de l'environnement car elle devient une variable interne. De plus, comme l'enrichissement change le type de la variable, on perd une variable d'environnement. Le nombre d'environnements est réduit ainsi que le nombre de lignes de la matrice.

Exemple 8.1.4 – Prenons pour exemple la transformation potentielle d'une matrice donnée, voir table 8.2 à gauche pour l'ajout d'un multiplexe régulateur sur la variable d'environnement O .

Env		Phénotypes	
<i>O</i>	<i>R</i>	<i>I</i>	<i>M</i>
0	0	osc(0-1)	
0	1	td(1)	td(0)
1	0	td(2)	
1	1		td(0)

Env	Phénotypes		
<i>R</i>	<i>I</i>	<i>M</i>	<i>O</i>
0	td(2)		osc
1	td(1)	td(0)	

Table 8.2 – Matrice de validation avant (gauche) et après (droite) enrichissement d'un multiplexe sur la variable d'environnement *O*. En vert sont les cases qui nécessitent une attention particulière. Les significations des motifs « osc » et « osc(0-1) » sont disponibles chapitre 5 section 5.3.2.

Dans ce cas la colonne *O* de la partie « Env » du tableau doit être supprimée. En supprimant cette colonne on constate qu'il ne reste plus que *R* comme variable d'environnement. Le nombre de ligne est donc divisé par 2. A l'opposé une nouvelle variable interne *O* apparaît : la variable *O* mène à une nouvelle colonne dans la matrice. L'ensemble des connaissances sur les phénotypes de cette variable est donc à ajouter à la nouvelle matrice. Les comportements des autres variables doivent aussi être modifiés. Dans cette exemple on constate que, pour une même valeur de *R* ($R = 0$) deux comportements différents de *I* (cases en vert dans la matrice) sont possibles : la différence est due à l'état de *O*. Pour construire la matrice associée à l'enrichissement, il faudra donc décider quel comportement on souhaite atteindre lorsque $R = 0$, vu qu'il n'y plus qu'une seule case associée (en vert dans le tableau de droite). Si l'on ne peut pas se prononcer, alors on laisse une case vide, mais par défaut on choisit parmi les comportements définis dans la matrice avant enrichissement. Dans notre cas choisissons « td(2) ». Dans tous les autres cas qui ne posent pas problème nous pouvons reporter les anciens comportements dans la nouvelle matrice donnée dans la table 8.2 de droite.

L'ajout d'un multiplexe sur une variable d'environnement est principalement rencontré lorsque l'on décide d'insérer une variable. En effet, pour simplifier les étapes d'enrichissement, nous faisons souvent le choix de définir l'ajout d'une variable étape par étape en commençant par la créer en tant que variable d'environnement avec l'ensemble des régulations qu'elle implique.

8.2 Insérer une variable

Passons maintenant à la méthodologie à suivre pour insérer une variable qui va faire intervenir l'ajout d'un multiplexe.

8.2.1 Motivation biologique d'un tel enrichissement

Le modélisateur cherche à ajouter une variable lorsque il constate qu'un élément abstrait est absent du graphe de régulation et nécessite de disposer d'un marqueur complémentaire. Il peut aussi être intéressant de vouloir ajouter une variable pour expliciter un marqueur abstrait dans le modèle, dont les comportements sont connus. L'ajout d'une variable permet de gagner en expressivité du modèle car en plus d'ajouter des transitions possibles, il augmente le nombre d'états composant le graphe de transition. Partant de l'élément à formaliser, on identifie les cibles de l'élément en question, et le type de régulation (coopération avec d'autres éléments, inhibition, activation, *etc.*) et finalement on identifie les régulations de ses cibles et ces propres régulation.

Prenons une version précédente du modèle de *Pseudomonas aeruginosa* publiée dans un article de 2004 [Bernot et al., 2004] où le calcium n’est pas présenté, dont le graphe est représenté en figure 8.4. Le calcium joue un rôle dans l’activation de la virulence de la bactérie ; on souhaiterait donc enrichir le modèle pour le rendre cohérent avec les connaissances actuelles [King et al., 2020].

8.2.2 Modification du graphe de régulation

Comme pour l’insertion d’un multiplexe la première étape est d’incorporer les informations précitées dans le graphe de régulation. On commence par ajouter la variable en définissant son domaine et le sens qu’elle va prendre. Ensuite on spécifie tous les liens que la variable va avoir avec le système. En premier lieu on va lier la variable par les éléments qu’elle régule (régulations en aval), puis dans un second temps on s’intéresse aux régulations du système sur la variable (régulations en amont de la variable). Pour représenter l’ensemble des régulations on va donc *a priori* devoir ajouter des multiplexes.

Exemple 8.2.1 – Dans notre exemple nous allons ajouter au graphe de régulation proposé figure 8.4, le calcium avec pour domaine $\llbracket 0, 1 \rrbracket$, où 0 et 1 correspondront respectivement à une absence et présence du calcium.

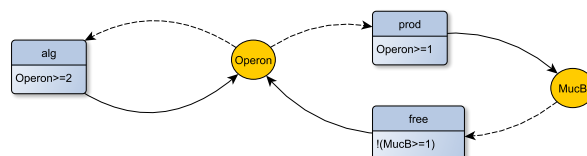


Figure 8.4 – Graphe de régulation de production de mucus *Pseudomonas aeruginosa* avant enrichissement par le Calcium

On s’intéresse dans un premier temps aux régulations en aval de la nouvelle variable **Calcium**. Pour chaque nouvelle régulation nous nous fions à la méthode d’ajout d’un multiplexe. Dans le cadre du système de production du mucus de *Pseudomonas aeruginosa*, on sait que le calcium régule directement et positivement l’opéron. On peut donc ajouter un multiplexe **ca** qui représente l’activation du calcium sur la variable **opéron** (Graphe de régulation enrichi présenté en figure 8.1, page 87). Ici, comme le calcium est un élément extra-cellulaire il n’y aura pas de régulation en amont de la variable.

Calcium sera donc une variable d’environnement. Pour la suite, il y aura quelques différences notables (dans l’élaboration de la matrice par exemple) entre l’ajout d’une variable environnementale et une variable interne au modèle.

De plus, ici le multiplexe ajouté est simple car il ne tient compte que de la variable ajoutée. Nous verrons dans la dernière partie du manuscrit, chapitre 10, lors de l’ajout de la variable **ROS** que les multiplexes ajoutés formalisent souvent une coopération de plusieurs entités du système.

8.2.3 Modification cohérente des paramètres

L’ajout d’une variable induit généralement l’insertion de nouveaux multiplexes qui lient la variable au réseau représenté : le nombre de paramètres est donc modifié. Si la variable ajoutée

est une variable interne, on commence par regarder les paramètres en aval de la variable, puis on travaillera sur les paramètres de la variable elle-même. Enfin, tant que le sens biologique des ressources d'un paramètre n'est pas modifié par l'enrichissement, la valeur du paramètre peut être conservée.

Exemple 8.2.2 – Ici, l'ajout de la variable **Calcium** augmente de 1 le nombre de prédécesseurs de la variable **operon** et par conséquent le nombre de paramètres de cette dernière double (voir table 8.3 ci-dessous).

<i>Koperon</i>	
<i>Koperon</i>	<i>Koperon, ca</i>
<i>Koperon, free</i>	<i>Koperon, free ca</i>
<i>Koperon, alg</i>	<i>Koperon, alg ca</i>
<i>Koperon, alg free</i>	<i>Koperon, alg free ca</i>

Table 8.3 – Table des paramètres d'**operon** après ajout de la variable **Calcium**. En vert sont les paramètres ajoutés par la présence de calcium comme ressource potentielle.

Avant de savoir que le calcium pouvait avoir un impact sur la régulation de la production du mucus chez *Pseudomonas aeruginosa* le système était étudié *in vitro* dans un milieu sans calcium. On en déduit que les paramètres avant ajout du calcium (autrement dit sans la ressource calcium) peuvent être conservés. Il reste ensuite à étudier les paramètres de l'opéron lorsque le calcium est présent. Quand le calcium est présent, il est capable de forcer l'activation de l'**opéron** à son maximum; donc $K_{operon, ca} = 2$ et par les conditions de Snoussi, il vient : $K_{operon, free ca} = 2$, $K_{operon, prod ca} = 2$ et $K_{operon, free prod ca} = 2$.

Dans certains cas l'estimation des paramètres à la main est impossible et *TotemBioNet* estimera s'il y a des paramétrisations cohérentes.

8.2.4 Non régression et révision cohérente de la matrice de validation

Après la modification des paramètres associés au graphe vient la modification de la matrice de validation. L'ampleur de la modification de la matrice dépend du type de la variable ajoutée.

Si c'est une variable interne on ajoute simplement une colonne à la matrice préexistante dans la catégorie phénotype. Le nombre de ligne de la matrice n'est donc pas modifié. Cette colonne sert à spécifier le comportement de la nouvelle variable. Pour chacune des cases on essaiera à l'aide des connaissances biologiques de spécifier un comportement consensus. Le reste de la matrice reste sensiblement identique car les autres comportements n'ont pas de raison d'être modifiés.

Si en revanche nous venons d'ajouter une variable d'environnement, on ajoute une colonne à l'environnement. Le nombre de lignes d'environnement va donc être augmenté (voir définition d'un environnement chapitre 5). Il sera multiplié par la taille du domaine de la variable ajoutée. Il faut d'abord déterminer les comportements observés avant ajout qui peuvent être reportés. Enfin, pour finir la nouvelle matrice de validation on formalisera, s'il est connu, le comportement pour l'ensemble des cases restantes des lignes nouvellement créés.

Exemple 8.2.3 – **Calcium** est une variable booléenne d’environnement. La matrice enrichie aura donc 1×2 lignes, puisque la matrice précédente n’avait qu’une ligne, cf table 8.4 . On reporte alors les anciens comportements pour la non-régression. Nous avons précédemment indiqué que l’ensemble des abstractions pour créer le modèle initial ont été faites en absence de calcium. Les comportements présentés dans la matrice du modèle initial, dans la table 8.4 peuvent donc être reportés sur la première ligne de la matrice de la table 8.5 (qui correspond au milieu sans calcium). A ce stade la matrice de non régression est terminée car nous avons reporté les spécifications provenant du modèle précédent. Nous n’avons plus qu’à codifier les possibles nouveaux comportements (en fond vert) qui enrichiront la validation dans le cadre des nouvelles conditions d’environnement.

Phénotypes	
Operon	MucB
$((\text{Operon} = 2) \Rightarrow A'G((\text{Operon} = 2)))$	
$((\text{Operon} < 2) \Rightarrow A'G((\text{Operon} < 2)))$	

Table 8.4 – Matrice de validation pour le modèle de *Pseudomonas aeruginosa* avant enrichissement avec le **Calcium**.

Environnements	Phénotypes	
Calcium	Operon	MucB
0	$((\text{Operon} = 2) \Rightarrow A'G((\text{Operon} = 2)))$ $((\text{Operon} < 2) \Rightarrow A'G((\text{Operon} < 2)))$	
1	td(2)	

Table 8.5 – Matrice de validation pour le modèle de *Pseudomonas aeruginosa* après enrichissement avec le **Calcium**. Le fond vert indique les enrichissements faisant partie de la nouvelle matrice enrichie. En blanc les comportements reportés de la matrice d’origine.

Dans le cas général, le placement correct des comportements hérités peut être complexe. Il faut considérer l’état global du système avant insertion de la variable pour déterminer où placer les comportements hérités. Maintenant que nous avons passé en revue les enrichissements par ajout d’entités dans le graphe de régulation, nous allons décrire à l’ajout d’un niveau à une variable préexistante.

8.3 Ajouter un niveau à une variable préexistante

8.3.1 Motivation biologique d’un tel enrichissement

Cet enrichissement est souvent utilisé lorsque l’on cherche à gagner en expressivité. Augmenter le nombre d’états et de transitions pour une variable apporte des nuances de niveau qui permettent de modéliser de nouveaux événements avec finesse.

Exemple 8.3.1 – Le réseau de régulation de *Pseudomonas aeruginosa* avait à l’origine été décrit sous forme booléenne [Bernot et al., 2004], avec un graphe de régulation présenté en figure 8.5.

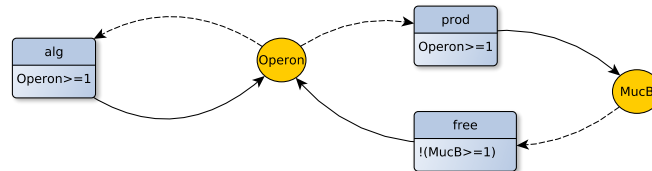


Figure 8.5 – Graphe de régulation du modèle booléen de la production de mucus chez *Pseudomonas aeruginosa*

Les auteurs ont constaté que l’expressivité dynamique du modèle était restreinte. On ne peut pas faire en sorte que la boucle négative et la boucle positive soient toutes deux fonctionnelles de manière concomitante, car le graphe de transitions ne possède pas assez d’états. Pour pouvoir ajouter de l’expressivité on choisit, dans un premier temps, la variable à laquelle on va ajouter un seuil. Il faut évaluer le sens biologique des nouveaux seuils de cette dernière, et l’intervalle dans lequel on souhaite ajouter le nouveau seuil.

Pour obtenir l’expressivité que l’on a aujourd’hui sur le modèle, nous devons ajouter un niveau à la variable *operon*. En effet, les boucles positives et négatives sont en compétition au même seuil dans le modèle vis à vis de la variable *operon*, mais dans le système biologique, le seuil d’activation de *prod* sera différent de celui de l’auto-activation de l’opéron : c’est donc le domaine de cette variable qu’il va falloir augmenter pour rendre les deux boucles fonctionnelles.

8.3.2 Modification du graphe de régulation

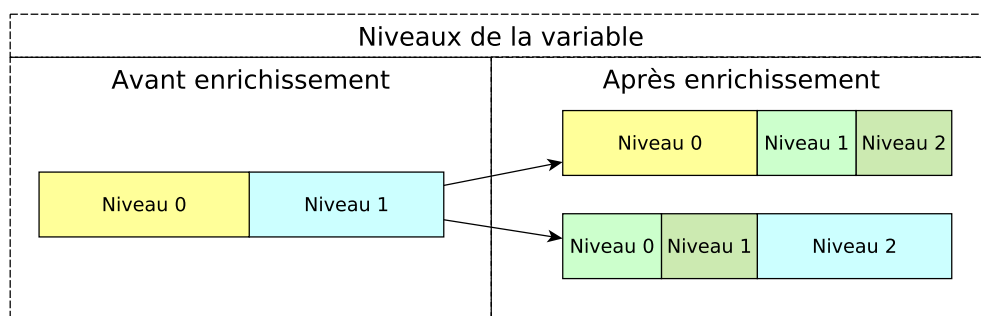


Figure 8.6 – Représentation des évolutions possibles des précédents niveaux après les différents enrichissements possibles. En jaune le niveau 0 avant enrichissement, en bleu le niveau 1 avant enrichissement. Après enrichissement le sens du niveau non dédoublé reste identique mais la valeur associée peut être amenée à changer.

Il n’y a pas de modification de la variable. Par contre il faut réévaluer les formules des multiplexes en aval de cette variable. Pour cela il faut connaître l’intervalle initial de la variable

que l'on va diviser pour obtenir de nouveaux niveaux abstraits. Deux choix s'offrent à nous dans l'exemple : soit on divise le niveau 0 en deux, soit on divise le niveau 1 en deux.

Division du niveau 1. Si on divise en deux le niveau 1, on va simplement s'intéresser aux multiplexes qui impliquent la variable d'intérêt au niveau 1. En effet le niveau 0 ne changeant pas de sens, les régulations qui y sont associées n'ont pas à être modifiées. En revanche, pour chaque mention du niveau 1 dans un multiplexe (chaque atome de la forme *operon* ≥ 1), il faut préciser si l'on fait référence à la partie basse de l'ancien intervalle 1 ou à sa partie haute : voir figure 8.6.

Exemple 8.3.2 – Dans le cas du modèle booléen de *Pseudomonas aeruginosa*, on se concentre sur les multiplexes (**prod** et **free**) qui régulent d'autres variables en aval et qui contiennent dans leurs formule *operon* ≥ 1 . Pour chacun de ces multiplexes il faut spécifier si l'on reste au niveau 1 ou si l'on monte au niveau 2. Le choix dépend des connaissances que l'on a. Dans notre cas on va considérer que l'auto-activation de l'opéron ne peut se faire qu'à partir d'un seuil relativement grand de molécules à l'inverse de l'activation de *MucB* : le multiplexe **prod** conserve la même formule (*operon* ≥ 1) et le multiplexe **Alg** est modifié (*operon* ≥ 2). Finalement, on obtient le graphe de régulation de la figure 8.1, page 87.

Division du niveau 0. Si on divise le niveau 0, alors tous les multiplexes qui nécessitaient d'avoir la variable au niveau 1 s'activent maintenant au niveau 2. En effet, le niveau 0 étant dédoublé en niveau 0 et 1 : l'ancien niveau 1 devient donc le niveau 2 après enrichissement voir figure 8.6. L'ajout d'un seuil au sein de l'intervalle 0 n'impacte aucun multiplexe car $x \geq 0$ n'est pas un atome autorisé (puisque toujours vrai). Par conséquent ce choix de seuil 0 n'est qu'une étape préparatoire qui induira la création d'au moins un nouveau multiplexe (section 8.1).

8.3.3 Modification cohérente des paramètres

L'ajout d'un niveau à une variable implique la réévaluation des paramètres touchés par le dédoublement de l'intervalle.

Exemple 8.3.3 – Dans notre exemple nous avons dédoublé le niveau 1. Tous les paramètres associés à la variable qui ont pour valeur 1 devront donc être réévalués à l'aide d'expériences de pensée et pourront maintenant prendre 1 ou 2 comme valeur. A l'inverse, les paramètres qui avaient pour valeur 0 resteront à 0. Voir les tables 8.7 et 8.6.

\mathcal{K}_{operon}
$K_{operon} = 0$
$K_{operon, free} = 1$
$K_{operon, alg} = 1$
$K_{operon, alg free} = 1$

Table 8.6 – Table des paramètres d'*operon* pour le modèle *Pseudomonas aeruginosa* booléen. En vert les paramètres à réévaluer.

\mathcal{K}_{operon}
$K_{operon} = 0$
$K_{operon, free} = 1$
$K_{operon, alg} = 1$
$K_{operon, alg free} = 2$

Table 8.7 – Table des paramètres d'*operon* pour le modèle *Pseudomonas aeruginosa* multivalué.

8.3.4 Non régression et révision cohérente de la matrice de validation

Dans la matrice de validation : les comportements des variables non impliquées par le dédoublement d'un niveau n'ont pas de raison d'être modifiées, seuls ceux de la variable avec des niveaux dédoublés sont à revoir. Pour chaque environnement, le comportement indiqué dans l'ancienne matrice pourra être conservé s'il ne concerne pas l'intervalle scindé. Dans les autres cas, on va devoir a minima préciser dans les formules le nouveau niveau concerné. Cela peut conduire à modifier la formule, voir à la supprimer si il n'y a plus de consensus biologique.

Exemple 8.3.4 – Pour notre exemple du passage du modèle booléen vers une version multivaluée, on conserve entièrement la colonne qui indique les comportements de la variable *MucB*, mais pas celle de *operon* dont le nombre de niveaux été modifié.

Environnement	Phénotypes	
<i>Calcium</i>	<i>Operon</i>	<i>MucB</i>
0	$((Operon = 1) \Rightarrow A'G((Operon = 1)))$ $((Operon < 1) \Rightarrow A'G((Operon < 1)))$	
1	td(0)	td(1)

Table 8.8 – Exemple fictif de matrice de validation pour le modèle de *Pseudomonas aeruginosa* booléen. En vert le comportement qui devra être adapté pour le passage du modèle booléen au multivalué.

Prenons une matrice théorique représentant des comportements attendus du modèle booléen table 8.8 et regardons les comportements de la variable *operon* que l'on peut conserver. Le comportement « td(0) »¹, traduit que la variable opéron doit tendre vers son niveau minium. Sachant que nous avons dédoublé le niveau 1, le sens abstrait du niveau 0 n'a pas été modifié. Un comportement asymptotique vers le niveau 0 n'est donc pas remis en cause par l'enrichissement et peut donc être conservé.

A l'inverse lorsque l'on rencontre un comportement qui est dépendant du niveau qui é été dédoublé alors le comportement doit être réévalué, c'est le cas notamment pour le phénotype associé à la formule :

1. Ce comportement a été ajouté artificiellement pour l'exemple, il ne fait pas parti de la matrice « réelle » du modèle booléen.

$$\begin{aligned} (((Operon = 1) \Rightarrow A'G((Operon = 1))) \\ ((Operon < 1) \Rightarrow A'G((Operon < 1)))) \end{aligned}$$

Nous devons donc estimer si le niveau 1 présent dans la matrice non enrichie correspond au niveau 1 ou 2 pour la matrice enrichie. Biologiquement, cette formule traduit le fait que lorsque une bactérie est mucoïde (*operon*=1), elle le reste et, inversement, lorsque elle ne l'est pas (*operon*=0) elle ne le devient jamais. D'après l'enrichissement, le niveau mucoïde correspond au niveau 2 de l'opéron. Cela indique que tous les niveaux 1 doivent être transformés en niveau 2. La formule de la nouvelle matrice devient donc (voir table 8.9) :

$$\begin{aligned} (((Operon = 2) \Rightarrow A'G((Operon = 2))) \\ ((Operon < 2) \Rightarrow A'G((Operon < 2)))) \end{aligned}$$

Environnement	Phénotypes	
<i>Calcium</i>	<i>Operon</i>	<i>MucB</i>
0	$\begin{aligned} (((Operon = 2) \Rightarrow A'G((Operon = 2))) \\ ((Operon < 2) \Rightarrow A'G((Operon < 2)))) \end{aligned}$	
1	td(0)	td(1)

Table 8.9 – Matrice de validation pour le modèle de *Pseudomonas aeruginosa* multivalué. En vert clair le comportement modifié adapté à l'ajout d'un de niveau de la variable *operon*.

Remarque – Cette section traite du dédoublement d'un niveau d'une variable mais il est bien sûr aisé de généraliser à l'ajout d'un nombre quelconque de seuils lors d'un unique enrichissement.

Augmenter le nombre de niveaux de certaines variables permet de gagner en expressivité et en précision sur la dynamique du modèle, mais il existe un autre moyen de gagner de la précision dans un modèle, qui peut s'avérer très utile pour la spécification d'un modèle générique d'un système biologique à une problématique particulière : scinder une variable en deux variables complémentaires.

8.4 Scinder une variable ou un multiplexe préexistant

8.4.1 Motivation biologique d'un tel enrichissement

Notre formalisme de modélisation permet un grand nombre de variantes dans les modèles car il offre une grande souplesse d'abstraction. Lorsque l'on cherche à représenter par exemple la coopération de plusieurs voies métaboliques, on peut être amenés à représenter plusieurs éléments biologiques par une même variable. Dans le cadre de la conception d'un modèle par enrichissement il est possible que certaines abstractions du modèle deviennent trop grossières pour traiter une nouvelle question biologique. Par exemple de nombreux modèles ont été développés pour représenter les régulations entre populations d'un même écosystème. Un modèle abstrait initial pourrait représenter, à l'aide de 3 variables, des plantes, des herbivores et des carnivores. Supposons que nous voulions travailler sur l'évolution d'un herbivore particulier en fonction de l'ensemble de ses prédateurs dans l'écosystème. On comprend que la variable herbivore doit être scindée au moins en deux catégories : l'herbivore d'intérêt et les autres herbivores. Cet enrichissement permettrait d'avoir un suivi comportemental de la population d'herbivores.

La motivation pour scinder un multiplexe en plusieurs repose sur les mêmes motivations biologiques.

Exemple 8.4.1 – Supposons que nous souhaitions enrichir le modèle de *Pseudomonas aeruginosa* avec de nouveaux mécanismes capables de réguler la quantité d’alginate (*AlgU*). Dans le modèle initial (figure 8.1), cette protéine est abstraite dans la variable **operon** car elle suit les mêmes évolutions². Pour que nous puissions représenter correctement ces nouvelles régulations il faut dans un premier temps modifier le graphe de régulations pour expliciter *AlgU* sous forme de variable. En effet, sans variable, il nous est impossible de représenter des nouvelles régulations spécifiques à *AlgU*. Nous devons donc prendre le parti de scinder la variable **operon**.

8.4.2 Modification du graphe de régulation

Commençons comme pour tous les enrichissements, par la modification du graphe de régulation. Pour scinder la variable choisie, il faut d’abord déterminer le nombre de variables qui remplaceront dans l’enrichissement la variable initiale. Par défaut on prendra le même nombre de niveaux pour chacune de ces nouvelles variables filles que de niveau de la variable mère.

Exemple 8.4.2 – Nous avons décidé de scinder la variable mère **operon** en deux variables filles : **operon** et **AlgU**. **operon** fille ne représentera plus que l’élément génétique et **AlgU** la quantité de protéine *AlgU*. **operon** fille représentera le niveau d’activation de l’opéron. **AlgU** représente une quantité de la protéine *AlgU*.

Une fois les variables filles déterminées, nous allons leur assigner les régulations de la variable mère dans le graphe enrichi. Il est possible qu’une même régulation concerne plus d’une variable fille : de nouvelles régulations peuvent donc être créées. La modification du graphe de régulation peut donc être importante. Pour simplifier la démarche, on traite les régulations en amont et en aval séparément.

8.4.2.1 Assignation des régulations en aval de la variable mère

Nous listons l’ensemble des régulations en aval de la variable mère, puis nous associons chacune de ces régulations à une/ou plusieurs variable(s) fille(s) à l’aide des connaissances biologiques. Cette étape permet de définir les modifications des multiplexes et des arcs en aval.

Exemple 8.4.3 – Toutes les régulations sortantes de **operon** mère ont été listées (colonne de gauche dans le tableau 8.10). La première régulation $\rightarrow prod$ représente l’activation de la production de la protéine *MucB*. Biologiquement, c’est l’expression d’un gène présent sur l’opéron qui permet la production d’une protéine : la source de la régulation mère sera donc remplacée par la variable **operon** fille. C’est pourquoi nous avons indiqué la régulation dans la colonne associée à **operon** fille du tableau 8.10. Toujours pour la même raison la régulation $\rightarrow alg$ dans le graphe enrichi gardera pour source l’opéron (**operon** fille). Nous avons fait l’inventaire de toutes les régulations sortantes de la variable mère et aucune n’a été assignée à la variable fille **AlgU**, d’où les cases vides pour la colonne **AlgU** du tableau 8.10.

2. Pour rappel, la production d’*AlgU* est directement due à l’activation de l’opéron tout comme *Muc*. ces deux protéines sont co-exprimées

	variables mères	Variables filles	
Régulations en aval des variables	Operon	Operon	AlgU
	→ prod	→ prod	
	→ alg	→ alg	

Table 8.10 – Table représentant les régulations *en sortie* de la variable mère et l’association de chacune d’entre elles avec les variables filles. « $\rightarrow prod$ » de la colonne **operon** mère indique la régulation de **opéron** dans le graphe de régulation initial qui régule *via* le multiplexe **prod** la variable **MucB**. La partie gauche du tableau indique la régulation dans le graphe initial et la partie de droite du tableau (après la double barre verticale) indique les variables filles et leurs régulations en aval associées.

En suivant les indications de la table 8.10, les arcs et multiplexes sortant des variables filles ont été dessinés figure 8.7, page 100. En conséquence, ces multiplexes sortants ne sont donc plus reliés à la même variable abstraite ; leurs formules doivent être redéfinies. Lorsque parmi les variables filles, il y a une variable qui joue le rôle de la variable mère dans les multiplexes en aval, la modification est simple car seul le nom de la variable mère est remplacé par le nom de la variable fille qui joue le même rôle. Si la signification du multiplexe change, alors il faut se référer à la section 8.1 qui indique les étapes à suivre pour la modification ou l’ajout d’un multiplexe. Enfin, le nom du multiplexe peut être revu lors de l’enrichissement pour une meilleure compréhension du modèle.

Exemple 8.4.4 – Ici la variable fille **operon** dans notre exemple porte le même nom que la variable mère, le nom dans les formules des multiplexes nouvellement associés à la variable fille ne seront pas modifiés : les formules de multiplexes **prod** et **alg** restent donc inchangées. Pour plus de clarté il est possible de renommer les multiplexes en aval des nouvelles variables. c’est pourquoi nous renommons le multiplexe **alg** par **prod2** puisque ce n’est pas l’alginate qui a une action sur lui même mais l’opéron (**prod2**).

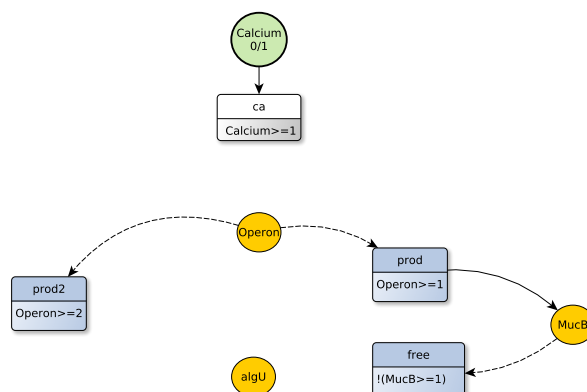


Figure 8.7 – Graphe de régulation en cours de construction avec les variables filles et leurs régulations en aval

Remarque – Si dans le tableau une régulation en aval de la variable mère est assigné à plus d’une variable fille alors la formule du multiplexe doit être modifiés en conséquence et prendre en compte les deux variables filles.

Cette première étape permet de construire en partie le nouveau graphe de régulations en y ajoutant les nouvelles variables et leurs régulations en aval.

Exemple 8.4.5 – Pour notre exemple, nous obtenons le graphe présenté figure 8.7. Dans ce graphe seules les sorties des variables filles sont dessinées. Chaque formule des multiplexes sortants a été adaptée à la nouvelle nomenclature des variables filles.

8.4.2.2 Assignation des régulations en amont de la variable mère

Comme l’on vient de le faire pour les régulations en aval, nous allons transférer chaque régulation ayant pour cible la variable mère vers les variables filles. Pour cela on édite un tableau comme à l’étape précédente. Cette fois les multiplexes sont sources des régulations à considérer.

Exemple 8.4.6 – La protéine *MucB* est capable de se lier à la protéine *AlgU* la rendant inactive. Dans le modèle initial, cette régulation était représentée par la régulation **free** → *operon*. On peut donc en déduire que la régulation sera, dans le graphe enrichi, appliquée à *AlgU*. De même nous savons que *MucB* et *AlgU* sont exprimés par l’opéron. La régulation **alg** → *operon* représente le contrôle de *algU* par l’opéron, sachant que le contrôle de *MucB* est assuré par le multiplexe **prod**. On renommera donc **alg** en **prod2**. Dans le graphe enrichi, il y aura donc un multiplexe qui régulera *algU*. Enfin, le calcium environnemental joue sur l’activation de l’élément génétique, la régulation **ca** → *operon* sera donc associée à la variable *operon* fille dans le graphe de régulation enrichi. Ces informations sont résumées dans le tableau 8.11.

Régulations en amont des variables	Variables mères	Variables filles	
		Operon	Operon
	free →		free →
	alg →		prod2 →
	ca →	ca →	

Table 8.11 – Régulations *en entrée* de la variable mère et des variables filles. A gauche la variable présente avant enrichissement et à droite les variables remplaçant la variable mère.

Remarque – Il est possible qu’une régulation en amont de la variable mère soit assignée à plus d’une variable fille. Dans ce cas, le multiplexe en amont de la variable mère aura dans le graphe de régulation enrichi plusieurs cibles (une par variable fille régulée).

A ce niveau le graphe de régulation enrichi peut donc être complété avec les régulations en amont des variables filles. On pourrait croire que la modification du graphe de régulation est terminée. Il n’en est rien : en explicitant de nouveaux concepts abstraits il faut réfléchir au lien exact entre ces nouveaux éléments ; et donc déterminer en plus s’il existe des régulations *entre* les variables filles.

Exemple 8.4.7 – Dans notre cas nous savons que l’accumulation de *AlgU* ne fait que forcer l’activation de l’opéron dès que la protéine est présente on peut donc ajouter une dernière régulation de *AlgU* vers *operon* via le multiplexe **alginate** et sa formule associée : $AlgU \geq 1$.

On finalise ainsi le graphe de régulation enrichi.

Exemple 8.4.8 – La figure ci-dessous montre le graphe de régulation enrichi final.

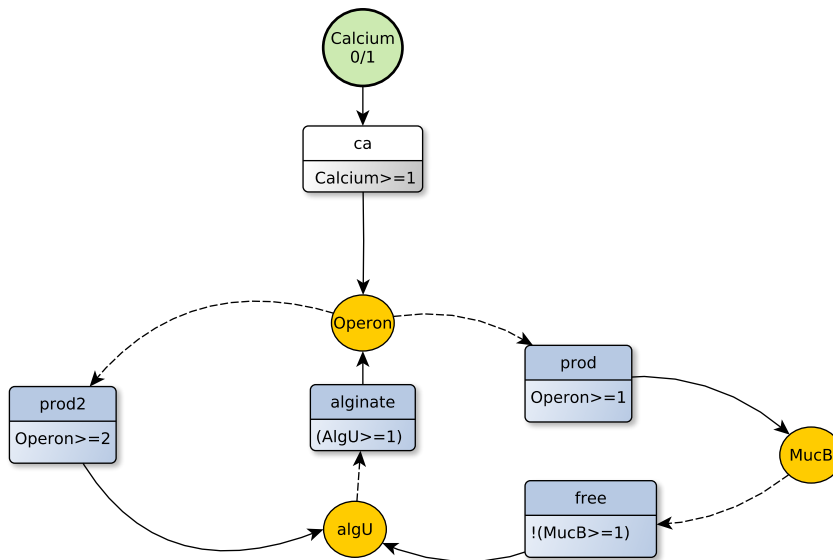


Figure 8.8 – Graphe de régulation en cours de construction avec les variables filles et leurs régulations en aval

L’ordre dans lequel on traite les régulations en amont et en aval est important. En effet, dans le cas particulier où l’on va scinder une variable qui possède une auto-régulation comme la variable **Operon**, l’ordre des opérations implique en premier la modification du multiplexe qui concernait l’auto-régulation (**alg**). Sa mise à jour entraîne nécessairement une réévaluation de la régulation en aval. Par conséquent, si dans les étapes pour scinder une variable on commençait par la modification des régulations en amont, il faudrait réévaluer tous les multiplexes de ce type une seconde fois.

8.4.3 Modification cohérente des paramètres

Comme la modification du graphe de régulation lorsque l’on scinde une variable est relativement conséquente, la modification des paramètres l’est aussi. On devra identifier, d’une part, les paramètres des nouvelles variables créées, et d’autre part réévaluer les paramètres des variables directement en aval des variables filles.

8.4.3.1 Réévaluation des paramètres des variables filles

Le nombre de paramètres à réévaluer dépend des régulations en amont de la variable mère transformées en plusieurs variables filles. Si aucune des régulations en amont de la variable mère n'est associée à plus d'une variable fille alors, le nombre de paramètres associé aux variables filles sera généralement plus faible que le nombre initial de paramètres associé à la variable mère. En effet, le nombre de régulations est identique, cependant ces dernières sont redistribués sur plusieurs variables et le nombre de prédécesseurs pour chaque variable est plus faible (sauf cas particuliers), ce qui réduit la combinatoire des ressources possibles pour chaque variable fille.

Exemple 8.4.9 – Supposons la table 8.12 théorique suivante qui résume la scission d'une variable mère en 3 variables filles où chaque régulation mère n'est associée qu'à une seule variable fille.

Régulations en amont des variables	Variables mères	Variables filles		
	Mère	Fille 1	Fille 2	Fille 3
	R1 →	R1 →		
	R2 →		R2 →	
	R3 →			R3 →

Table 8.12 – Régulations *en entrée* de la variable mère et des variables filles. À gauche la variable présente avant enrichissement et à droite les variables remplaçant la variable mère.

Avant enrichissement le nombre de paramètres pour la variable **Mère** est de $2^3 = 8$ alors que le nombre total de paramètres pour les trois variables filles (**Fille 1**, **Fille 2**, **Fille 3**) est de $3 \times 2^1 = 6$.

Dans ce cadre l'ensemble des régulations étant juste réarrangées, pour chaque nouveau paramètre associé à une variable fille, il existe un paramètre "équivalent" au sens abstrait, dans le modèle non enrichi associé à la variable mère. Il est donc possible de s'appuyer, dans ce cas, sur ces valeurs des paramètres avant enrichissement pour identifier les nouveaux paramètres.

Lorsque une régulation de la variable mère est associée à plus d'une variable fille alors de nouvelles régulations sont créées dans le graphe de régulation, et en conséquence de nouveaux paramètres sont ajoutés par rapport à un enrichissement sans ajout de régulation.

Enfin, des régulations peuvent aussi être ajoutés lorsque les variables filles ont des interactions entre elles. Dans ce cas l'identification des paramètres ne peut pas s'appuyer sur les paramètres du modèle non enrichi.

Continuons maintenant l'enrichissement commencé pour la division de la variable **Operon** du modèle de *Pseudomonas aeruginosa*.

Exemple 8.4.10 – En partant du tableau 8.11 des régulations en amont de la variable mère, on constate que le prédécesseur de la variable **Operon** fille (**ca**) existait déjà pour la variable **Operon** mère. Pour identifier le paramètre $K_{Operon, ca}$ du modèle enrichi on peut s'appuyer sur le paramètre $K_{Operon, ca}$ du modèle non enrichi car les conditions de régulations en amont que proposent ces deux paramètres sont les mêmes. Cependant il faut garder en tête que les conditions en aval de la variable **operon** fille sont probablement différentes que celles de la variable **operon** mère dans le graphe de régulation initial. Pour pouvoir reporter la valeur du paramètre initial, il faut aussi vérifier la cohérence du sens donné aux cibles en aval et vérifier que la valeur de l'ancien paramètre ne fait pas de contresens dans l'activation des cibles en aval.

8.4.3.2 Réévaluation des paramètres pour les variables en aval des variables filles

La réévaluation des paramètres des variables en aval des variables filles n'est pas systématique. Si les régulations en aval de la variable mère ont toutes été associées à une seule fille, aucune nouvelle régulation n'est créée. Le sens des régulations n'est donc pas modifié par le dédoublement de variable. Dans ce cas l'ensemble des valeurs de paramètres peut être conservé.

Reprenons le modèle dont le graphe enrichi est présenté figure 8.8. Le tableau 8.10 des régulations aval de la variable mère et leur transfert aux variables filles indique qu'aucune des régulations de la variable mère n'est associée à plus d'une variable fille. De plus le sens biologique (et donc la formule) des multiplexes associés n'a pas été modifié (comparaison entre Figures 8.1 et 8.8). Dans ce cas les paramètres de la totalité des variables directement en aval de la modification n'ont pas à être modifiés.

Exemple 8.4.11 – La régulation $\xrightarrow{\text{prod}}$ **MucB** avait pour paramètres dans le modèle non enrichi $K_{\text{MucB}} = 0$ et $K_{\text{MucB}, \text{prod}} = 1$. Le multiplexe **prod** le même sens et contient la même formule dans le graphe de régulations enrichi et non enrichi. Par défaut, comme ni le sens abstrait de **MucB** ni ses cibles n'ont été modifiés pendant l'enrichissement les valeur des paramètres de la variable **MucB** dans le modèle enrichi pourront être reportés.

A l'inverse si des régulations en sortie ont été « ajoutées » c'est à dire que plusieurs variables sont associées à la même régulation mère, alors :

1. il y a augmentation du nombre de paramètres pour la variable doublement régulée. Une partie des paramètres sera à estimer une nouvelle fois.
2. les paramètres des variables cibles qui ne sont impactées que par une variable fille pourront être conservés.

8.4.4 Non régression et révision cohérente de la matrice de validation

La scission d'une variable représente un travail important sur la structure du graphe de régulation, mais *a contrario* cela ne va pas impliquer de modifications drastiques de la matrice de validation. En effet, excepté pour les comportements des nouvelles variables créées, aucun comportement ne doit être modifié. Cela permet de conserver les propriétés du modèle avant enrichissement.

Les nouvelles variables créent de nouvelles colonnes dans la matrice et pour chaque environnement il faut déterminer le comportement cohérent avec les connaissances biologiques. Les comportements de la variable mère peuvent aider la détermination de celui des variables filles.

Exemple 8.4.12 – La matrice de validation avant la scission de la variable **operon** est donnée en table 8.11, et la matrice après scission donnée en table 8.13. La colonne concernant la variable non scindée **MucB** n'a pas à être modifiée. La scission de la variable **operon**, engendre la suppression de la colonnes **operon** mère et son remplacement par deux colonnes **operon** et **AlgU** en vert dans la matrice.

Le comportement de la variable mère **Operon** dans le graphe de régulation initial était spécifié par la formule :

$$\begin{aligned} (((Operon = 2) \Rightarrow A'G((Operon = 2))) \\ ((Operon < 2) \Rightarrow A'G((Operon < 2)))) \end{aligned}$$

Cette formule indique que si une bactérie est mucoïde elle le reste et inversement. Cependant, la mucoïdie de la bactérie est estimée à la quantité de mucus produite *via* la quantité de *AlgU*. On peut donc transférer ce comportement dans la case associée au même environnement pour la variable fille *AlgU* (voir table 8.13). Enfin, nous supposons que l'activation de l'opéron en absence de calcium n'est pas constante : le comportement de *Operon* est oscillant, on peut mettre « osc » dans la case correspondante (cf. table 8.13).

Environnement	Phénotypes		
<i>Calcium</i>	<i>Operon</i>	<i>AlgU</i>	<i>MucB</i>
0	osc	$\begin{aligned} &(((Operon = 2) \Rightarrow A'G((Operon = 2))) \\ &((Operon < 2) \Rightarrow A'G((Operon < 2)))) \end{aligned}$	
1	td(1)		

Table 8.13 – Matrice de validation pour le modèle de *Pseudomonas aeruginosa* après scission de la variable *Operon*.

Dans ce chapitre nous avons vu 4 types d'enrichissements : l'ajout d'un multiplexe, l'ajout d'une variable, l'ajout d'un niveau à une variable et le remplacement d'une variable par un ensemble de nouvelles variables. L'ajout d'un multiplexe est l'enrichissement le plus basique que l'on peut faire et, est très souvent réutilisé pour construire des enrichissements plus complexes tel que l'ajout d'une variable. Nous expliquons les démarches à suivre pour effectuer un enrichissement de manière méthodique. Nous avons illustré ces 4 types d'enrichissement sur le modèle de *Pseudomonas aeruginosa* qui est de petite taille. Cependant, tous les types d'enrichissement proposés ici peuvent être utilisés sur des modèles de plus grande taille, comme nous le verrons dans les chapitres 10 et 11, lorsque nous cherchons à raffiner le modèle générique de la régulation du métabolisme au cas d'une cellule cancéreuse agressive.

Conception incrémentale d'un modèle de régulation du métabolisme dans l'adénocarcinome ductal pancréatique

L'ensemble des notations pour les variables et multiplexes sont indiqués en annexe [A](#) pour simplifier la lecture en plusieurs étapes. De même l'ensemble des paramètres identifiés au cours de la conception incrémentale du modèle sont référencés en annexe [B](#).

Un modèle initial générique : régulation du métabolisme de cellules saines

Le modèle introduit dans ce chapitre se focalise sur la glycolyse en aérobie aussi appelée l'effet Warburg/Crabtree . Il a pour but de déterminer si cet effet peut être juste une conséquence des signaux de régulation classiques entre les principales voies du métabolisme central du carbone, ou si la glycolyse aérobie au contraire, repose sur des détails additionnels. Le modèle proposé repose sur le formalisme de René Thomas. Les différentes régulations y sont représentées qualitativement ce qui permet d'utiliser les approches formelles présentées précédemment, qui à leur tour aident le modélisateur à construire un modèle cohérent.

9.1	La régulation du métabolisme et la glycolyse en aérobie	108
9.1.1	Le métabolisme	108
9.1.2	Les acteurs du métabolisme	110
9.1.3	La glycolyse en aérobie	113
9.2	Composition du graphe de régulation	114
9.2.1	Description du sens biologique des variables	114
9.2.2	Description du sens biologique des multiplexes	118
9.3	Formalisation des phénotypes : la matrice de validation	121
9.3.1	Comportements en l'absence de lipides et d'oxygène	122
9.3.2	Comportements en l'absence de lipides mais avec apport d'oxygène	124
9.3.3	Comportements avec un apport en lipides mais sans oxygène	126
9.3.4	Comportements avec apport en lipides et en oxygène	126
9.4	Spécifications détaillées : identification manuelle des paramètres	127
9.4.1	Paramètres des variables représentant des voies métaboliques	127
9.4.2	Paramètres des variables représentant la biomasse	131
9.4.3	Paramètres des variables représentant des métabolites	132
9.5	Validation de modèles à l'aide de <i>TotemBioNet</i>	138

La conception de ce modèle générique a débuté en 2015 et avait pour but de répondre à la question biologique suivante : est-ce que les régulations classiques du métabolisme cellulaire permettent, à elles seules, d'expliquer l'effet Warburg/Crabtree ? Répondre à cette interrogation par l'intermédiaire d'un modèle formel, a nécessité de se placer à la bonne échelle biologique et de définir tous les éléments pertinents à cette échelle. Les premières sections de ce chapitre décrivent le métabolisme ainsi que l'effet Warburg/Crabtree (de nos jours plus couramment appelé la glycolyse en aérobie) d'un point de vue biologique. La seconde, troisième et quatrième sections formalisent les connaissances biologiques. La seconde section se focalise sur la description des différents éléments du graphe de régulation, les variables et les multiplexes, puis fait le lien avec la biologie en décrivant les entités biologiques que chacune des variables abstrait. La troisième section définit les phénotypes globaux pour l'ensemble des conditions environnementales (*via* une matrice de validation). La quatrième section décrit les expériences de pensée ayant permis d'identifier la majorité des paramètres dynamiques liés au graphe de régulation. Enfin, la dernière section commente l'étape de validation effectuée à l'aide de *GreenBioNet* une méthode de validation que nous avons intégrée à *TotemBioNet*.

Ce travail se fonde évidemment sur des travaux antérieurs. Rajeev Khoodeeram a d'abord proposé un premier modèle [Khoodeeram et al., 2016] puis une seconde version plus aboutie dans sa thèse [Khoodeeram, 2021]. Quand j'ai repris ces travaux, plusieurs points nécessitaient des améliorations comme la variable de la biomasse à séparer en deux variables, la réduction du nombre niveau d'oxygène, ou encore la modification de la gestion des lipides, qui, une fois faites, ont permis la publication d'un modèle abouti de la régulation du métabolisme [Gibart et al., 2021c] pouvant être utilisé de base pour être enrichi au contexte du cancer du pancréas.

9.1 La régulation du métabolisme et la glycolyse en aérobie

Pour appréhender la conception d'un modèle d'un système biologique il faut dans un premier temps se familiariser avec la biologie du système étudié. A travers les sous-sections proposées, nous décrivons ce que représente le métabolisme [Berg et al., 2002, Nelson and Cox, 2012, Koolman and Röhm, 2012] quels en sont les principaux acteurs puis nous présenterons les grandes lignes du très documenté phénomène appelé la glycolyse en aérobie [Molenaar et al., 2009, de Alteriis et al., 2018].

9.1.1 Le métabolisme

Le métabolisme correspond à l'ensemble de toutes les transformations bio-chimiques, décomposables en réactions simples, qui se produisent dans une cellule ou un organisme. Ces réactions sont regroupées en plusieurs voies métaboliques (une voie métabolique correspond donc à une suite de réactions bio-chimiques). Une grande partie des ces voies métaboliques se retrouve conservée dans les différentes espèces du règne vivant. Que l'on considère un animal, végétal ou micro-organisme tous ces organismes conservent l'ensemble des réactions chimiques des grandes voies métaboliques telles que la glycolyse.

On distingue deux types de voies métaboliques : les anaboliques, qui sont des réactions de biosynthèse des constituants cellulaires (autrement appelé biomasse), et d'autre part les voies cataboliques, qui sont des réactions de dégradation de ces constituants cellulaires en petites molécules pour en libérer l'énergie nécessaire au fonctionnement cellulaire. Catabolisme et anabolisme

sont en constant équilibre, entre la production et le renouvellement des constituants nécessaires au maintien cellulaire et à la production d'énergie.

Les différentes voies métaboliques quel que soit leur type (cataboliques et anaboliques) sont interconnectées *via* des molécules spécifiques (intermédiaires réactionnels tel que le pyruvate) formant des carrefours dans les voies métaboliques, qui sont très importants pour la régulation du métabolisme. Nous nous intéresserons ici au métabolisme au niveau cellulaire car la glycolyse en aérobie est visible au niveau cellulaire¹. Une représentation simplifiée des différentes voies métaboliques d'une cellule est donnée Figure 9.1.

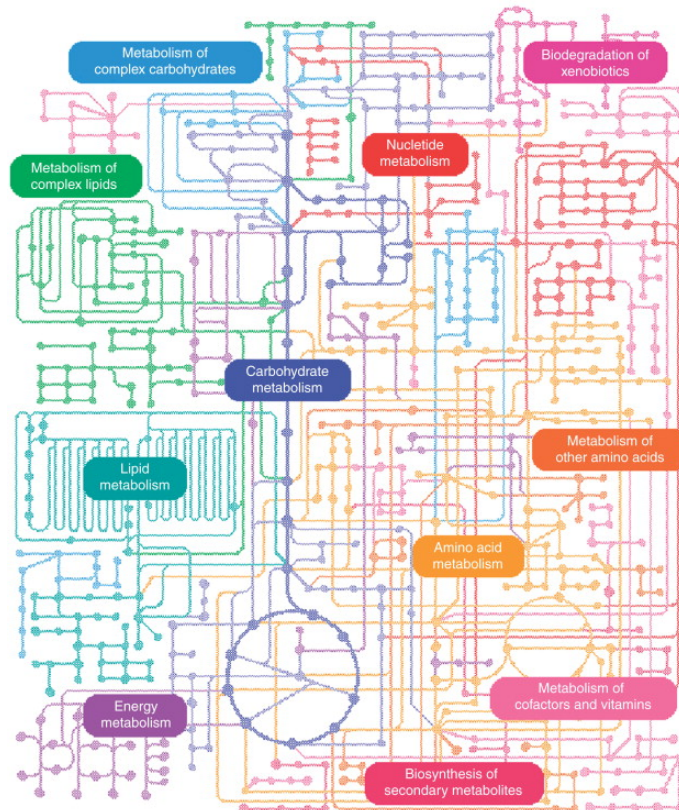


Figure 9.1 – Carte simplifiée des voies métaboliques issue de l'article [Boyer, 2004].

9.1.2 Les acteurs du métabolisme

L'état des cellules, comme la quiescence ou la prolifération, dépend de la régulation du métabolisme et principalement de l'activation des voies cataboliques ou anaboliques. Les voies cataboliques sont communément activées lorsque la cellule est en déficit d'énergie. A l'inverse, les voies anaboliques seront utilisées lorsque la cellule a un niveau énergétique suffisant. De plus, les voies anaboliques permettent la production de biomasse nécessaire pour le renouvellement des constituants cellulaires. Dans le métabolisme tout est donc une question d'équilibre entre ces types de voies pour la vie de la cellule.

9.1.2.1 Les acteurs du catabolisme

Lorsque la cellule est dans des conditions environnementales favorables [Yin et al., 2012] (présence de nutriments, sucre, acides aminés et oxygène), les voies cataboliques, dont les principales sont la glycolyse, la fermentation, le cycle de Krebs (ou cycle du citrate), la bêta-oxydation et la phosphorylation oxydative, vont se mettre en place pour produire de l'énergie sous forme de molécules d'*ATP* (adénosine tri-phosphate).

La description suivante reprend les réactions cataboliques dans l'ordre de dégradation en partant du glucose jusqu'à l'obtention d'énergie sous forme d'*ATP*.

1. Notez que dans le cas des micro-organismes, souvent l'organisme se confond avec le niveau cellulaire.

La glycolyse. Le glucose est transformé en pyruvate (métabolite central entre le cycle de Krebs et fermentation) *via* la glycolyse en 10 réactions chimiques dont 3 ne sont pas réversibles. Lors de ce processus, NAD^+ est un cofacteur nécessaire et sera oxydé en $NADH$. Ces deux molécules ont un noyau commun, le NAD (nicotinamide adénine dinucléotide). Le $NADH$ est ensuite utilisé pour générer de l'énergie sous forme d'ATP *via* le transfert de ses électrons, disponible dans la cellule pendant la phosphorylation oxydative (voie métabolique décrite plus loin dans cette section).

Après la première réaction au sein de la glycolyse (qui transforme le glucose en glucose-6-phosphate), il existe une bifurcation amenant les produits de cette réaction à la voie des Pentose Phosphate (PPP), connu pour être le producteur de briques élémentaires (acides aminés et nucléotides) pour la synthèse de biomasse (anabolisme). Dans notre modèle, la glycolyse est surtout considérée comme un producteur de pyruvate, de sorte que nous pouvons considérer qu'il y a une répartition constante du substrat (glucose) entre le PPP et la glycolyse, quel que soit le niveau d'apport en glucose. Le pyruvate peut ensuite intégrer deux voies différentes : soit il entre dans la voie de **la fermentation**, soit dans **le cycle de Krebs** ce qui est le cas le plus général dans les conditions environnementales classiques.

La fermentation. Le pyruvate accumulé *via* la glycolyse est réduit en lactate (cellule eucaryote) ou en éthanol (levure) par le lactate ou l'éthanol déshydrogénase. Ce processus permet la réduction du $NADH$ en NAD^+ nécessaire à nouveau pour le bon fonctionnement de la glycolyse et est une voie clef dans la glycolyse en aérobie. Généralement ce processus n'est actif qu'en absence d'oxygène.

Le cycle de Krebs. C'est une voie métabolique présente dans la mitochondrie voir (Figure 9.2) de la cellule qui contient 8 réactions et qui est ubiquitaire chez l'ensemble des "organismes aérobies" (organismes utilisant l'oxygène pour produire de l'énergie). Sa fonction première est d'oxyder les groupes acétyle du pyruvate et des composés lipidiques ou protéinés, pour récupérer l'énergie sous forme de $NADH$ ou $FADH_2$ ($FADH_2$ étant un cofacteur similaire au $NADH$). Ce cycle est très important à la fois dans les processus cataboliques mais aussi pour l'anabolisme si les réactions sont effectuées dans le sens inverse. L'alpha-cétoglutarate qui est un des substrats et produits du cycle est un élément qui permet une régulation efficace du cycle de Krebs.

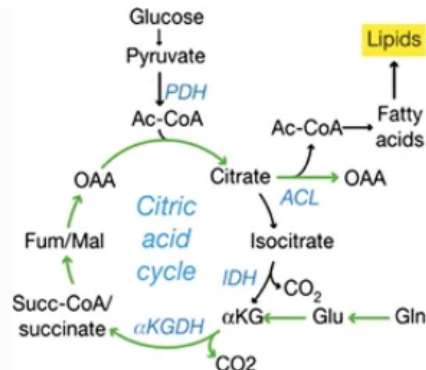


Figure 9.2 – Schéma des réactions du cycle de Krebs issu de l'article [Maus and Peters, 2017]

La phosphorylation oxydative. Si le cycle de Krebs est actif, la production de $NADH$ permet l'activation de la phosphorylation oxydative, voir Figure 9.2. Cette dernière est un processus de phosphorylation de l'ADP en ATP grâce à l'énergie libérée par l'oxydation de donneurs d'électrons ($NADH$) *via* la chaîne respiratoire. Au cours du processus de phosphorylation oxydative, des électrons sont transférés de molécules réductrices à des oxydants le long des différents complexes réalisant une cascade de réactions qui libèrent progressivement l'énergie provenant des

électrons. La dernière des réactions nécessite, comme précurseur, de l'oxygène qui est réduit en eau lors de la réaction. L'ensemble de l'énergie est utilisée par des pompes à protons pour générer un gradient électrochimique à travers une membrane biologique. Les protons s'accumulent d'un côté de la membrane, et leur reflux vers l'autre côté pour équilibrer le gradient de concentration entraîne un rotor moléculaire dans l'ATP synthase qui fournit l'énergie nécessaire à la conversion de l'ADP (Adénosine di-phosphate) en ATP. Cette dernière étape permet le stockage de l'énergie sous forme inerte pouvant ensuite être utilisée pour la création de macromolécules *via* la dégradation de l'ATP en ADP voire AMP (adénosine mono-phosphate).

Cycle de Krebs et phosphorylation oxydative sont deux mécanismes permettant la respiration en aérobie (respiration en présence d'oxygène).

La bêta-oxydation. Les lipides et les acides gras sont d'autres sources de carbone qui sont oxydés par la voie d'oxydation β , qui se déroule dans les mitochondries, fournisseur de *NADH* à la chaîne respiratoire.

9.1.2.2 Les acteurs de l'anabolisme

Les voies anaboliques ont pour but de créer de la biomasse. La biomasse représente l'ensemble des constituants de la cellule : les composés génétiques, protéiques ou encore lipidiques. L'anabolisme représente donc toutes les voies métaboliques permettant la construction des briques élémentaires de la cellule. Beaucoup plus éparses, les voies métaboliques pour la production de composés complexes sont regroupables par le type moléculaire qu'elles créent.

La production des composés génétiques. Les composés génétiques ou nucléotides nécessaires pour la formation de l'ADN (acide désoxyribonucléique) et l'ARN (acide ribonucléique) sont créés par la voie des pentoses phosphates (PPP). Outre la production de nucléotides, cette voie permet la formation d'acides aminés aromatiques. Cette voie est essentielle pour permettre la division cellulaire, principalement lors de la duplication de l'ADN. Elle permet aussi la régénération des composés génétiques contenus dans le noyau de la cellule.

La production des composés protéiques. Les protéines sont composées d'acides aminés, assimilés par l'alimentation ou synthétisés. La synthèse des acides aminés se fait par des voies métaboliques très différentes dans l'organisme car ils sont de nature très hétérogènes. Par exemple comme indiqué précédemment, les précurseurs de certains acides aminés aromatiques sont produits par la voie PPP, mais d'autres acides aminés comme la proline ou l'arginine peuvent être synthétisés par l'intermédiaire de l'alpha-cétoglutarate. Malgré l'hétérogénéité de la production des acides aminés (AA), la production finale des protéines associées est simple car elle lie par une liaison peptidique les acides aminés (cette réaction de liaison nécessite de l'énergie).

La production des composés lipidiques. La synthèse des lipides se fait par l'intermédiaire de la création d'acides gras (AG). Ces acides gras ont comme précurseur l'acétyl-coA également précurseur du cycle de Krebs. La synthèse *de novo* des acides gras est essentielle pour ensuite créer des lipides complexes comme les phospholipides ou le cholestérol nécessaire à la formation de la membrane cellulaire. La synthèse d'acide gras est permise par l'utilisation d'ATP et la condensation de molécules d'acétates. Pour former des acides gras plus long que 16 atomes de carbone, une

autre machinerie se met en place, comme la création d'insaturations sur les AG. Enfin, d'autres mécanismes permettent de coupler les lipides avec d'autres composés phosphorylés ou glycérolés.

L'activité des processus anaboliques et cataboliques présentés ci-dessus est très finement régulée à l'aide de molécules intermédiaires comme le pyruvate, l'acetyl-CoA, les ratios $ATP/(ADP + AMP)$ et $NADH/NAD^+$. En effet, leurs accumulation/déficit indique un certain état physiologique général de la cellule, ce qui lui permet, en réponse d'activer ou inhiber les différentes voies en fonction de ces besoins et signaux externes. La glycolyse en aérobie est une observation phénotypique d'un état cellulaire qui peut être considérée comme une réponse particulière d'une régulation du métabolisme à un environnement spécifique.

9.1.3 La glycolyse en aérobie

Dans des conditions normales, une cellule saine aura tendance à utiliser le processus de dégradation de macromolécules en passant par la glycolyse, cycle de Krebs pour produire 36 ATP par molécule de glucose *via* la phosphorylation oxydative. La fermentation n'est effective que si la phosphorylation oxydative est bloquée par l'absence d'oxygène et que le seul producteur d'ATP devient la glycolyse (à raison de 2 ATP par glucose). Dans ce second cas, l'activation de la fermentation évite que l'accumulation de pyruvate bloque la glycolyse et génère le nécessaire cofacteur NAD^+ . Le processus de phosphorylation oxydative est connu pour être très efficace pour produire de l'énergie mais relativement lent, tandis que le métabolisme fermentaire peut être très rapide malgré sa capacité de production faible d'ATP.

En passant de la rareté à l'abondance de nutriments (sucres et source de carbone), les cellules hautement prolifératives favorisent une dégradation rapide du glucose, ainsi elles favorisent la fermentation plutôt que la respiration, car cela peut offrir une production élevée d'ATP et de blocs de construction (acides aminés et nucléotides). Le passage de la respiration à la fermentation se produit même en présence d'oxygène. Ce phénotype glycolytique est connu sous le nom de glycolyse en aérobie aussi connu sous le nom de l'effet Warburg/Crabtree (ici nous n'aborderons pas la réversibilité de cet effet [de Alteriis et al., 2018]). Il s'agit d'une caractéristique des cellules à prolifération rapide, en particulier des cellules tumorales qui ont un accès rapide au glucose [Zhu and Thompson, 2019]. Beaucoup d'efforts ont été consacrés à déchiffrer le mécanisme d'action moléculaire qui sous-tend *la glycolyse en aérobie* [da Veiga Moreira et al., , Rigoulet et al., 2020].

Dans l'esprit de la description de haut niveau décrite dans cette section, nous avons conçu un modèle formel très abstrait de régulation du métabolisme des eucaryotes, comme le montre la Figure 9.3 car notre objectif est de représenter les principales interactions de haut niveau entre les voies métaboliques reproduire la glycolyse en aérobie pour constituer un modèle général solide et adaptable au contexte du cancer. Certains éléments moléculaires sont cruciaux (O_2 , $NADH/NAD^+$, ATP , ...) mais une description abstraite des voies majeures (glycolyse, Krebs, fermentation, ...) est en fait le niveau de description approprié par rapport à notre objectif, simplement parce que nous étudions les interactions entre les voies plutôt que le fonctionnement interne des voies elles-mêmes.

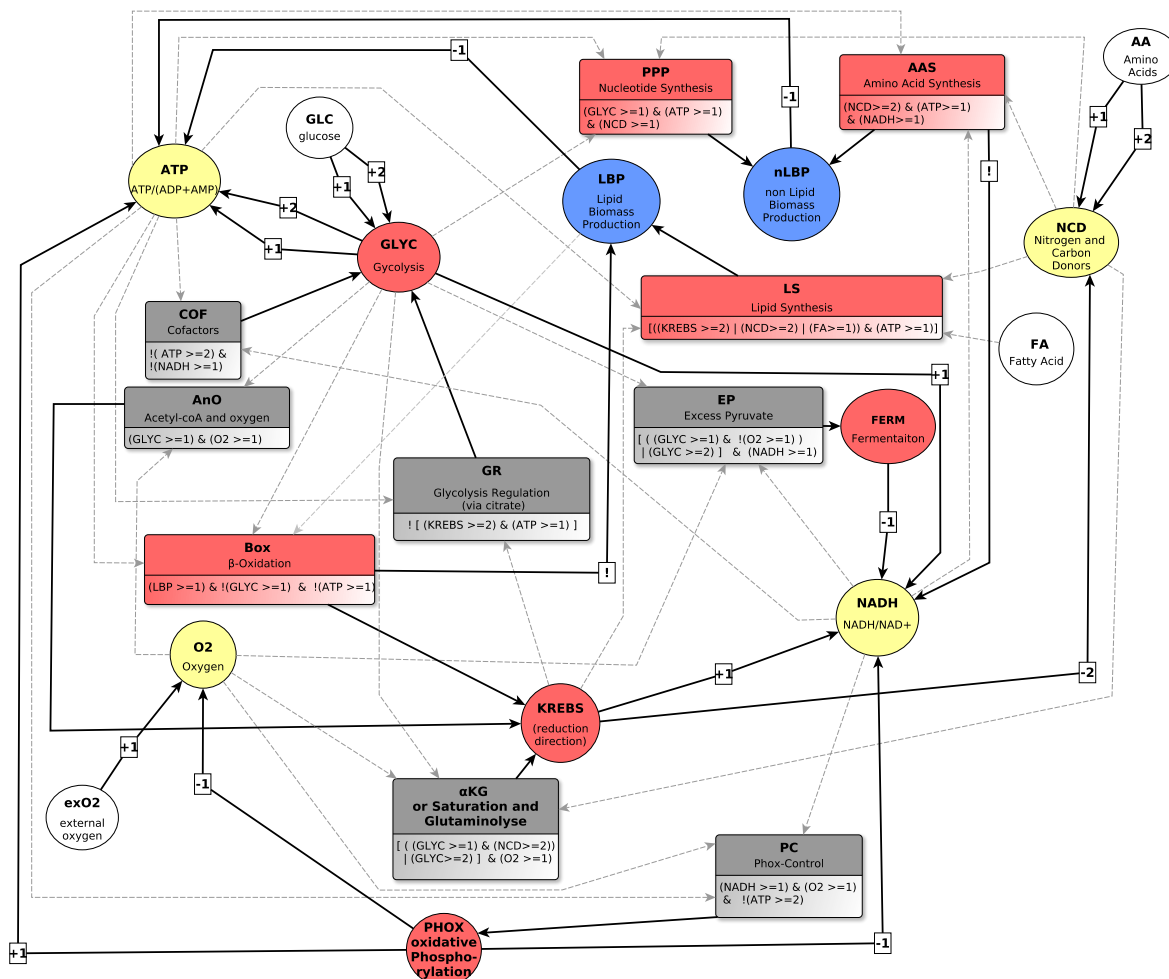


Figure 9.3 – Graphe de régulation du métabolisme de la cellule. Les cercles représentent des variables alors que les rectangles représentent des multiplexes. Les entités bleues font référence à la biomasse, les jaunes aux métabolites et les rouges aux voies métaboliques. Les lignes pleines indiquent les cibles de régulation. Les lignes pointillées désignent les sources de régulation (mathématiquement inutiles puisqu’elles peuvent être déduites de formules des multiplexes). Dans les formules logiques « ! » signifie la négation, « & » la conjonction et « | » la disjonction.

9.2 Composition du graphe de régulation

9.2.1 Description du sens biologique des variables

La description de chaque variable est essentiellement donnée pour justifier les arcs *sortants* (régulations) du graphe de régulation donné figure 9.3 mais aussi pour définir les seuils. Les variables sont séparées en quatre classes représentées par des couleurs différentes : blanc pour les nutriments environnementaux, jaune pour les métabolites d’échange, bleu pour la biomasse, et rouge pour les voies métaboliques. Dans cette section chaque variable décrite est soulignée et im-

médiatement suivie de l'intervalle de ses valeurs possibles. Cette typographie simplifiera par la suite le référencement et la lecture.

9.2.1.1 Les variables environnementales

L'apport nutritionnel de la cellule est représenté par 3 variables environnementales : **FA**, **GLC** et **AA**, représentant respectivement les Acides gras (ou fatty acids), les sucres (glucose), et les acides aminés. La dernière variable environnementale du graphe de régulation 9.3 est l'apport en oxygène représenté par la variable exO_2 .

- **exO_2 et FA** $\in [0, 1]$ Ces variables booléennes abstraient respectivement l'oxygène externe et les apports en lipides. Au niveau 1, **FA** peut participer aux apports lipidiques *via* le multiplexe **LS**, à condition que **ATP** soit présent. Au niveau 1, **exO₂** active l'oxygène dans la cellule par l'intermédiaire de **O₂**. Inversement, lorsque leur valeur est 0, l'apport en nutriments est *non suffisant* pour être utilisé normalement par la cellule, cela ne signifie pas pour autant qu'il n'y a aucun apport.

- **GLC et AA** $\in [0, 2]$ Ces variables à trois niveaux abstraient le glucose et les acides aminés (dérivés des protéines). Ce sont des nutriments cruciaux pour étudier l'effet Warburg/Cabtree [Liberti and Locasale, 2016, Dang, 2012]. En effet leur niveau de présence (concentration extracellulaire) influe sur l'activation de la glycolyse (**GLYC**) et les donneurs d'azote et de carbone. Un niveau intermédiaire pour ces variables est donc nécessaire pour pouvoir capturer au mieux les nuances comportementales des cellules. Ainsi, le niveau 0 représente une faible disponibilité de nutriment, conduisant à une activation négligeable des processus métaboliques en aval ; le niveau 1 représente une disponibilité « normale » suffisante pour les besoins de base du métabolisme, et le niveau 2 représente un niveau élevé de nutriments utilisés par la cellule induisant souvent une sur-activation de la glycolyse.

Finalement comme ces 4 variables correspondent à des apports nutritionnels, elle permettront de tester le modèle dans différents environnements. Chaque combinaison de valeurs pour les variables environnementales représente un contexte différent. Le nombre de contextes environnementaux est par conséquent de $2 \times 2 \times 3 \times 3 = 36$, et ils seront inventoriés de manière exhaustive à l'aide d'une *matrice de validation* dans la section 9.3.

9.2.1.2 Les variables représentant des métabolites

On comptabilise 4 variables qui représentent des métabolites ou ensemble de métabolites : **ATP**, **NADH**, **NCD** et **O₂**.

- **ATP** $\in [0, 2]$ représente le rapport de concentration de $ATP/(ADP + AMP)$. Elle abstrait le niveau énergétique de la cellule. Au cours des réactions anaboliques, l'**ATP** est transformé en **ADP** ou en **AMP** pour libérer de l'énergie. L'**ADP** et l'**AMP** peuvent ensuite être régénérés au cours de la glycolyse et/ou par la respiration mitochondriale *via* l'ATP synthase. Comme il existe des navettes entre le cytoplasme et la mitochondrie, le rapport $ATP/(ADP + AMP)$, en tant que ratio, peut être considéré comme presque homogène dans les différents compartiments de la cellule, même si individuellement le niveau d'**ATP** ou d'**ADP + AMP** varie fortement entre la mitochondrie et le cytoplasme. Pour cette variable, et toutes celles qui représentent, un ratio

notons que l'interprétation d'un niveau peut être double. En effet le niveau 2 pour l' ATP ($ATP=2$) peut indiquer soit un niveau élevé d'intensité de l'énergie c'est à dire une grande concentration en ATP , soit une faible concentration de molécules d' $ADP + AMP$. On retrouve le même principe quand la variable ATP est égale à 0 : soit l'énergie cellulaire est très basse (concentration en ATP est faible), soit la concentration d' $ADP + AMP$ est très élevée. Enfin, $ATP=1$ représente un certain équilibre moléculaire pour le ratio $ATP/(ADP + AMP)$.

○ $NADH \in [0, 1]$ Ici comme pour la variable ATP , $NADH$ représente le rapport de concentration moyen de $NADH/NAD^+$, $NADPH/NADP^+$ ainsi que $FAD/FADH_2$. Ces derniers appartiennent aux ensembles de molécules qui sont des cofacteurs transporteurs d'électrons importants pour le bon fonctionnement de certaines réactions chimiques. Le choix de modélisation consistant à abstraire ensemble ces trois types de rapports mérite quelques commentaires. En effet, le couple $NADH/NAD^+$ est couramment utilisé dans les processus cataboliques alors que le couple $NADPH/NADP^+$ est surtout utilisé dans les processus anaboliques. De plus, dans la figure 9.3, la glycolyse active le rapport $NADH/NAD^+$ (flèche « +1 ») et elle ne s'adresse pas au même rapport que la phosphorylation oxydative, qui agit sur $NADPH/NADP^+$. Néanmoins, selon notre niveau de description qui est très abstrait :

- A notre granularité de résolution temporelle, nous considérons que leurs valeurs moyennes ont la même signature (profil d'évolution au cours du temps) comme observée montrée par exemple dans le contexte du cycle cellulaire [Moreira et al., 2016].
- Les seuils entre 0 et 1 sont purement symboliques dans le formalisme de Thomas de sorte qu'il n'est pas nécessaire que les seuils $NADH/NAD^+$ et $NADPH/NADP^+$ soient dans la réalité quantitativement égaux. Le même raisonnement s'applique au quotient $FAD/FADH_2$. Par conséquent, nous n'avons aucune raison valable ni de distinguer entre ces trois ratios ni de distinguer plus de deux niveaux différents pour les cibles $NADH$ (principe de parcimonie [Sober, 2015]). Finalement, le niveau 0 représente simplement un niveau trop faible pour agir sur ses cibles (faible $NADH$ ou, également, fort NAD^+), contrairement au niveau 1 (et ceci s'applique aussi pour les autres ratios).

○ $NCD \in [0, 2]$ NCD pour Nitrogène and carbon donors, représente les dérivées des acides aminés donneurs d'azote et de carbone, utiles à la cellule. Ces éléments sont utilisés pour alimenter certaines voies métaboliques comme la voie des pentoses phosphates ou des processus anaboliques comme le cycle de Krebs à l'envers (représenté par le (αKG)). Au niveau 0, l'action de NCD est trop faible pour participer à l'activation d'un processus métabolique; au niveau 1, la variable participe à l'activation de la voie des pentose-phosphate (PPP), et au niveau 2, elle permet la production de αKG , via (αKG) par glutaminolyse même en l'absence de O_2 et il participe également à la production acides aminés et à la synthèse de lipides via AAS et LS .

○ $O_2 \in [0, 1]$ représente la concentration intracellulaire en oxygène. L'oxygène participe activement à la production d' ATP via la phosphorylation oxydative ($phox$). et à l'inverse l'hypoxie (faible taux d'oxygène cellulaire) est un signe d'alarme porteur de régulation pour certains processus comme le cycle de Krebs ou la glycolyse. Les cibles de la variable O_2 ne dépendent que de l'absence ou la présence d'oxygène, deux niveaux 0 et 1 sont donc suffisants pour représenter ces régulations.

Les métabolites explicités dans cette section permettent de réguler les différentes voies métaboliques de la cellule. La prochaine section est dédiée à la présentation des variables des dites voies métaboliques.

9.2.1.3 Les variables représentant des voies métaboliques

On décompte 4 variables décrivant les voies métaboliques dans le graphe de régulation : **GLYC**, **FERM**, **KREBS** et **PHOX**.

- **GLYC** $\in [0, 2]$ représente la glycolyse [Axelrod, 1967] qui produit de l'ATP (**GLYC** active **ATP** dans le graphe) et du pyruvate à partir de glucose et de NAD^+ . Trois de ces réactions sont limitantes et réalisées à l'aide de différentes enzymes, telles que la Phospho-Fructo-Kinase (PFK) qui a un rôle majeur dans la régulation de la glycolyse. Le niveau 0 représente une glycolyse qui ne produit pas assez d'intermédiaires réactionnels ou d'ATP ou de pyruvate utiles à d'autres processus métaboliques (comme le cycle de Krebs). Le niveau 1 représente un flux glycolytique suffisant pour activer les processus métaboliques qui y sont reliés : (PPP², Krebs, etc. ainsi que la fermentation en hypoxie). Enfin, le niveau 2 représente un niveau d'activité élevé où la glycolyse peut être considérée comme en sur-fonctionnement, par rapport aux besoins d'une cellule saine et ce, sous des conditions nutritives optimales. Dans ce cas, la glycolyse favorise l'accumulation de pyruvate mais également le processus de fermentation si il y a suffisamment de NADH, même en présence d'oxygène (phénotype retrouvé pour les cellules à forte division).

- **FERM** $\in [0, 1]$ représente la fermentation. Cette voie métabolique devient importante lorsque l'apport en oxygène n'est plus suffisant. Comme la production de $NADH$ est la seule cible de **FERM** dans le modèle, **FERM** est une variable booléenne. Le niveau 1 représente alors une activité de la fermentation suffisamment importante pour régénérer le NADH, tandis que le niveau 0 ne le permet pas.

- **Krebs** $\in [0, 2]$ représente le cycle de Krebs. *Il ne représente pas les réactions inverses* (branches réductrices du cycle de Krebs) qui sont implicitement prises en compte dans certains multiplexes. Ce cycle central dans le métabolisme permet l'oxydation de groupes des molécules résultant de différents processus cataboliques (glycolyse, β -oxydation, dégradation des acides aminés...). Un flux suffisant sur cette voie permettra à la cellule de réaliser les différentes oxydations du cycle et par conséquent de réduire efficacement le NAD^+ en $NADH$ [Anderson et al., 2018] (**KREBS** active **NADH**). Son flux dépend du niveau d'oxygène cellulaire, mais également de la quantité de précurseurs disponibles [Lowenstein, 1967]. Le niveau 0 représente un faible flux qui ne permet pas d'obtenir de façon notable $NADH$. Le niveau 1 représente un flux du cycle de Krebs capable de réduire NAD^+ en $NADH$ provenant des processus cataboliques de base (glycolyse et β -oxydation). C'est le flux normal pour des cellules saines avec un apport adapté de nutriments. Au niveau 2, le cycle de Krebs est en sur-fonctionnement et alerte la cellule pour qu'elle baisse l'activité des processus cataboliques, comme la glycolyse (via **GR**) et favorise les processus anaboliques comme la synthèse des lipides (via **LS**).

- **PHOX** $\in [0, 1]$ représente la phosphorylation oxydative, une voie métabolique dont les enzymes sont présentes dans la mitochondrie. Tout en consommant de l'oxygène et du $NADH$ (**PHOX** inhibe **NADH** et **O2**) cette voie permet la création d'ATP via l'ATP synthase (**PHOX** active **ATP**). Plusieurs réactions chimiques permettent la réduction d'une molécule d'oxygène en une molécule d'eau. Ces étapes génèrent un gradient électrochimique porteur d'énergie, qui est

2. En termes de *flux*, la PPP peut être considérée comme un concurrent de la glycolyse. Mais en termes de *régulation*, parce que la variable **GLYC** extrait tous les métabolites intermédiaires impliqués dans la glycolyse, **GLYC** est un activateur de **PPP** (par l'intermédiaire d'un de ses métabolites précoces).

utilisé pour former de l'*ATP*. Le fonctionnement de cette voie dépend entièrement de la quantité d'oxygène [Green and MacLennan, 1967]. *PHOX* active toutes ses cibles au même seuil, il s'agit donc d'une variable booléenne.

Il reste une dernière catégorie de variables à présenter : les variables liées à la biomasse.

9.2.1.4 Les variables représentant la biomasse

◦ $LBP \in [0, 1]$ représente l'abstraction de l'ensemble des voies de synthèse et le stockage de lipides (complexes ou non). Les lipides sont utiles pour le renouvellement des composants ainsi que la prolifération de la cellule. La production de biomasse consomme de l'énergie : *LBP* inhibe *ATP*. En outre, la production de biomasse lipidique *LBP* peut participer à l'activation de Krebs par le biais de *Box*. Quel que soit le type de biomasse, il y a soit production et stockage soit aucun des deux. Ce sont des variables booléennes.

◦ $nLBP \in [0, 1]$ représente la production et le stockage de biomasse non lipidique composés de protéines et d'éléments génétiques (ADN et ARN) utiles à la division cellulaire et au renouvellement des constituants cellulaires. Comme pour la production de biomasse lipidique la production de biomasse non lipidique (e.g. exemple la production de protéine) nécessite de l'*ATP*.

Notons que la synthèse des lipides, est un processus qui consomme *ATP*. Techniquement, comme *LS* est un multiplexe, on délègue la consommation d'*ATP* à la variable *LBP*. Il en est de même pour *nLBP*.

Passons maintenant à la description des multiplexes du graphe de régulation du métabolisme.

9.2.2 Description du sens biologique des multiplexes

Dans la figure 9.3, les multiplexes gris représentent les régulations du métabolisme, alors que les rouges abstraient des voies métaboliques. A chaque multiplexe est associée une formule logique, exprimant les conditions sous lesquelles la régulation (pour les multiplexes gris) ou la voie métabolique (pour les multiplexes rouges) est active. Notons que lorsqu'une régulation se fait par l'intermédiaire d'une seule variable, aucune formule n'est associée au multiplexe explicitement ; on note simplement le niveau de régulation auquel elle s'applique. C'est le cas par exemple pour les régulations qu'exerce *GLC* sur *GLYC*. D'un point de vue formel cette formule de multiplexe existe³. C'est un atome de la forme $v \geq n$ ou $v \neq n$ qui n'est pas représenté simplement par soucis de lisibilité du graphe. De plus ces multiplexes ne sont pas explicités ci-dessous car cette action directe de la variable source de l'arc est décrite en section 9.2.1..

Dans cette section, les multiplexes sont triés en fonction de la variable qu'ils régulent et, pour faciliter la lecture, chacun des multiplexes est souligné et immédiatement suivi de sa formule associée.

3. elle est définie dans le processus de validation via *TotemBioNet*

9.2.2.1 Multiplexes régulant des métabolites

Les régulateurs du *NADH* :

□ **AAS** $(NCD \geq 2) \& (ATP \geq 1) \& (NADH \geq 1)$ Ce multiplexe représente les processus de production anaboliques d'acides aminés (Amino Acid Synthesis) qui favorisent la production de biomasse non lipidique sous forme protéique. Ils sont principalement synthétisés à partir d'autres acides aminés collectés par la cellule dans son environnement extérieur. Le multiplexe **AAS** résume l'ensemble des éléments nécessaires pour produire de nouveaux acides aminés, comme l'azote et le carbone. Ces deux derniers éléments sont récupérés des produits de dégradation des acides aminés internalisés par la cellule et sont donc représentés dans le modèle par la variable *NCD* ce qui justifie $(NCD \geq 2)$ dans la formule du multiplexe. De plus pour la production d'acide aminés les réactions nécessitent une grande quantité de *NADH* ($NADH \geq 1$), additionné à de l'*ATP* pour certaines d'entre elles ($ATP \geq 1$). La conjonction de ces trois conditions aboutit à la formule de ce multiplexe [Kumagai, 2013, Reitzer, 2009].

De plus, *NADH* étant consommé, **AAS** inhibe *NADH* : c'est encodé par le « ! » (négation) sur la flèche sortante de **AAS** vers *NADH*.

9.2.2.2 Multiplexes régulant les voies métaboliques

Les régulateurs de *GLYC* :

□ **COF** $!(ATP \geq 2) \& !(NADH \geq 1)$ représente les cofacteurs nécessaires à un déroulement correct de la glycolyse : *ADP* et *NAD*⁺ doivent avoir des concentrations suffisamment élevées. Cela signifie que $ATP < 2$ et $NADH < 1$, comme déjà expliqué dans la section 9.2.1.2. Ainsi, la formule de **COF** formalise correctement la régulation qu'exerce les cofacteurs sur la glycolyse.

□ **GR** $![(Krebs \geq 2) \& (ATP \geq 1)]$ représente les inhibiteurs du flux glycolytique comme une absence de l'enzyme PFK (Phospho-fructo-kinase), et une accumulation de citrate. L'enzyme PFK de la glycolyse est allostériquement inhibée par l'*ATP* [Mansour, 1963]. Ainsi la *ATP* participe à une inhibition de la variable *GLYC* via la PFK.

De plus, le produit final de la glycolyse alimente le cycle TCA par sa transformation en citrate (via l'acétyl-coa), qui, en cas d'excès, inhibe la glycolyse. Rappelons que le TCA est abstrait par **KREBS** et qu'un excès de citrate est produit quand $Krebs \geq 2$. Au bilan *GLYC* est inhibé si les conditions de l'*ATP* et du citrate sont réunies : $(Krebs \geq 2) \& (ATP \geq 1)$. La négation « ! » au début de la formule indique l'inhibition.

Les régulateurs de *KREBS* :

□ **AnO** $(GLYC \geq 1) \& (O_2 \geq 1)$ représente l'action de l'Acétyl-coA en tant que principal précurseur du cycle de Krebs : il dérive directement à partir du pyruvate (produit de la glycolyse) [Lowenstein, 1967], ce qui est formalisé comme $GLYC \geq 1$. De plus, l'accumulation d'acétyl-coA provenant de la glycolyse active le cycle de Krebs si l'oxygène est présent ($O_2 \geq 1$) [Shi and Tu, 2015].

□ **Box** $(LBP \geq 1) \& !(GLYC \geq 1) \& !(ATP \geq 1)$ représente β -oxydation qui est une voie métabolique représentant la dégradation des acides gras en Acétyl-CoA [Houten and Wanders, 2010]. L'oxydation est effectuée quand le niveau d'énergie de la cellule est relativement faible ($ATP \geq 1$), lorsque la réserve de lipides stockés est assez haute ($LBP \geq 1$) et quand la glycolyse

n'est pas suffisamment efficace pour produire de l'énergie $!(GLYC \geq 1)$ [Lowenstein, 1967]. La formule du multiplexe **Box** est donc la conjonction de ces conditions.

□ **α KG** $[(GLYC \geq 1) \& (NCD \geq 2) \mid (GLYC \geq 2)] \& (O2 \geq 1)$ représente l'accumulation d' α -cétoglutarate. C'est un métabolite et aussi un intermédiaire réactionnel du cycle de krebs (voir figure 9.2). On retrouve cette accumulation en présence d'oxygène $O2 \geq 1$ et lorsque le cycle de Krebs a un flux très important provoqué par la saturation de la glycolyse ($GLYC \geq 2$) [Wu et al., 2016]. Une seconde manière d'accumuler de l' α -cétoglutarate est de passer par la glutaminolyse [McKeehan, 1986, Yang et al., 2017]. On retrouve ce processus lorsque l'apport en acides aminés sous forme d'azote et de carbone est important ($NCD \geq 2$) lorsque la glycolyse fonctionne normalement ($GLYC \geq 1$) en présence d'oxygène. Finalement, on obtient la formule $[(GLYC \geq 1) \& (NCD \geq 2) \& (O2 \geq 1)] \mid [(GLYC \geq 2) \& (O2 \geq 1)]$ que l'on peut ensuite factoriser pour obtenir la formule présentée en début de paragraphe.

Les régulateurs de **FERM** :

□ **EP** $(NADH \geq 1) \& [(GLYC \geq 2) \mid ((GLYC \geq 1) \& !(O2 \geq 1))]$ représente l'excès de pyruvate, qui active le flux de la fermentation. La fermentation a besoin de NADH comme cofacteur ($NADH \geq 1$). Elle peut être activée lorsqu'il y a surproduction de pyruvate *via* la glycolyse ($GLYC \geq 2$), mais aussi lorsque en absence d'oxygène $!(O2 \geq 1)$ nous avons une production normale de pyruvate *via* la glycolyse ($GLYC \geq 1$).

Les régulateurs de **PHOX** :

□ **PC** $(NADH \geq 1) \& (O2 \geq 1) \& !(ATP \geq 2)$ représente le contrôle de **PHOX** (phosphorylation oxydative). La phosphorylation oxydative nécessite un apport d'oxygène ($O2 \geq 1$) comme précurseur de la chaîne. Il nécessite également du NADH ($NADH \geq 1$). De plus, cette voie n'est activée que lorsque le réservoir d'énergie de la cellule n'est pas trop élevé [Wilson, 2017], c'est-à-dire $!(ATP \geq 2)$.

9.2.2.3 Multiplexes régulant la biomasse

Les régulateurs de **nLBP** :

□ **PPP** $(GLYC \geq 1) \& (ATP \geq 1) \& (NCD \geq 1)$ représente la production de nucléotides *via* la voie des Pentose Phosphate. C'est une voie qui favorise la production de biomasse non lipidique (**nLBP**) en produisant des nucléotides, qui sont des précurseurs de composants génétiques (AND/ARN). Cette voie produit du *NADPH*, qui est directement consommé pour produire de la biomasse non lipidique, de sorte que le résultat final de la production de *NADPH* est neutre pour le ratio $NAD^+/NADPH$ abstrait dans la variable **NADH**. En conséquence, nous ne mettons pas d'arc de régulation de **PPP** vers **NADH** dans le graphe (contrairement à **AAS**). La glycolyse est nécessaire à l'activation *via* le glycéraldéhyde-3-phosphate (G3P) ($GLYC \geq 1$) de **PPP**. Le Glycéraldéhyde-3-phosphate (G3P) est un intermédiaire réactionnel de la glycolyse et aussi le précurseur de la voie des phosphates de pentose. En outre, ces réactions endergoniques nécessitent un apport de carbone ($NCD \geq 1$) provenant de l'internalisation des acides aminés, ainsi que de l'ATP ($ATP \geq 1$) [Gregg and Prchal, 2018]. Finalement la formule représente donc la conjonction des trois éléments précités.

Le multiplexe **AAS** (Amino acids Synthesis) est aussi un activateur de **nLBP**. Il a déjà été décrit comme inhibiteur de la **NADH**, le contenu de la formule reste identique $(NCD \geq 2) \& (ATP \geq 1) \& (NADH \geq 1)$ car les voies de systhèse abstraites seront identiques.

les Régulateurs de **LBP** :

□ **LS (Lipids Synthesis)** $(((Krebs \geq 2) | (NCD \geq 2) | (FA \geq 1)) \& (ATP \geq 1))$ représente spécifiquement la synthèse des lipides composés d'acides gras. La synthèse des lipides utilise de l'énergie ($ATP \geq 1$) et des acides gras qui peuvent provenir directement de l'environnement cellulaire ($FA \geq 1$) ou par la voie de la synthèse des acides gras, dont le principal précurseur est l'acétyl-CoA. Ce dernier est à son tour fourni par le cycle de Krebs lorsqu'il est en sur-fonctionnement ($Krebs \geq 2$) ou par l'intermédiaire de la glutaminolyse directement dérivé de la dégradation des acides aminés ($NCD \geq 2$) [Stein and Stein, 1967]. Additionné à l'énergie donnée par l'ATP un seul des trois autres composants est nécessaire pour produire des lipides d'où l'utilisation des disjonctions entre les éléments.

Le multiplexe **Box** (β -oxidation) a déjà été décrit comme régulateur de **KREBS**, mais il permet aussi la dégradation des lipides. Il est par conséquent un inhibiteur de **LBP**, encodé dans la Figure 9.3 par le signe de négation « ! » de **Box** à **LBP**.

Suivant le cycle de vie en « V » indiqué dans l'introduction de la partie 2 du manuscrit, nous restons toujours dans la partie de la spécification des besoins du modèle mais nous passons maintenant à la formalisation des phénotypes, en mettant en place la matrice de validation.

9.3 Formalisation des phénotypes : la matrice de validation

La matrice de validation donnée dans la table 9.1 liste tous les éléments formalisés dans le graphe de régulation et comportements (phénotypes) du métabolisme, dans les limites du niveau d'abstraction choisi, en fonction de différents contextes environnementaux. Ces contextes sont des conditions nutritives différentes (Section 9.2.1.1). Chaque ligne représente un ensemble de valeurs pour les variables environnementales (variables blanches dans la figure 9.3, définies section 9.2.1.1). Il y a 36 combinaisons pour les variables d'environnements, donc la matrice de validation contient 36 lignes. Chaque colonne représente une variable dont le comportement est biologiquement interprétable, éventuellement observable expérimentalement. En plus des 4 variables environnementales (colonnes gris claires) qui définissent les différents environnements, il existe 10 variables systémiques internes dont le comportement peut être connu (cercles jaunes, rouges et bleus dans la Figure 9.3). Pour définir l'ensemble des comportements phénotypiques attendus dans les différents environnements nous avons utilisé les macros formalisées dans le chapitre 4. Pour rappel, les principales macros utilisées sont « td » pour tend vers et « osc » pour définir une oscillation (voir sous-section 5.3.1 et 5.3.2). Pour l'oscillation, il est possible dans le cas où la variable n'est pas booléenne, d'osciller de différentes manières. Dans ce cas si le comportement oscillatoire est connu avec précision, alors les bornes sont indiquées entre parenthèses à la suite du mot « osc » dans le tableau (par exemple ligne 14 osc(0-1) pour la **GLYC**). Dans le cas où l'on sait que l'on ne doit pas atteindre un certain comportement on ajoutera la négation « ! » (exemple de !tds(0) pour **FERM** ligne 4). Toujours concernant la formalisation des comportements, il est

important de noter que nous n'utilisons que la sémantique *Fair-Path CTL* et, par conséquent, chaque comportement indiqué dans la matrice de validation doit être compris comme *asymptotique*. Enfin, chaque case grise indique l'absence d'un consensus pour représenter le phénotype. Cela ne sous entend pas forcément qu'il n'y a aucune connaissance sur un type particulier de cellule, mais seulement que le comportement attendu n'est pas général dans les différents types cellulaires.

Les sections 9.3.1 à 9.3.3 ci-dessous listent les 36 contextes environnementaux de la matrice de validation. Parfois, deux contextes sont présentés dans la même ligne, par exemple les contextes 2 et 3, car les connaissances générales des comportements dans ces deux contextes sont identiques. Pour chaque ligne du Tableau 9.1 l'utilisation des différentes macros est justifiée à l'aide des connaissances générales en biologie. Graphiquement la flèche « \rightsquigarrow » suivie du numéro de la ligne est utilisée pour présenter chaque description de ligne. Sauf indication contraire, le terme « connaissances générales » fait référence aux livres de biochimie, tels que [Nelson and Cox, 2012, Koolman and Röhm, 2012, Berg et al., 2002].

Notons que dans aucune des descriptions apportées ci-dessous nous ne discutons du comportement de la variable *NCD*. Dans les faits, le comportement de la variable est directement dépendant de l'apport en acides aminés, et ce, qu'il y ait ou non consommation de composés carbonés ou azotés par d'autres mécanismes (comme par exemple la voie des pentose phosphates). En effet, la consommation ne permet jamais de faire baisser suffisamment la quantité de *NCD* pour que l'on passe le seuil. Pour simplifier, nous avons donc fait le choix de ne pas expliciter cette observation pour chaque ligne puisque la colonne *NCD* n'est qu'une copie de la colonne *AA*. Cependant, il existe quelques exceptions où l'on déroge à cette observation : ce sont les lignes qui indiquent que le phénotype général de la cellule n'est pas connu (lignes \rightsquigarrow 2, 3, 11, 12, 20, 21, 29 et 30). Pour plus de lisibilité nous avons fait le choix arbitraire d'organiser la description des comportements par l'état de l'apport en lipides et oxygène.

9.3.1 Comportements en l'absence de lipides et d'oxygène

On va donc commencer par définir les comportements pour tous les environnement qui se trouvent en l'absence de lipides et d'oxygène.

\rightsquigarrow **1** – *FA=0, exO2=0, GLC=0, AA=0* : les cellules ont besoin de suffisamment de nutriments pour répondre aux exigences métaboliques de la croissance et de la division cellulaire. Lorsqu'aucun nutriment n'est disponible, aucune voie métabolique ne fonctionne : chaque variable tend vers 0. Cela conduit à la mort cellulaire.

\rightsquigarrow **2 & 3** – *FA=0, exO2=0, GLC=0, AA=(1 or 2)* : nous n'avons pas à disposition un phénotype consensus. La cellule ne dispose que d'acides aminés comme source de nutriments. Il n'y a pas de connaissances pour affirmer que l'apport d'acides aminés seul est capable de provoquer une quelconque activité métabolique dépendante du carbone dans la cellule sauf pour les cellules cancéreuses en culture in vitro [Dang, 2012, Rigoulet et al., 2020]. Nous considérons donc que le destin des cellules est inconnu, sauf pour l'oxygène car sans apport la cellule se retrouve en hypoxie et *O2* tend vers 0.

\rightsquigarrow **4** – *FA=0, exO2=0, GLC=1, AA=0* : sans oxygène et avec du glucose comme unique source de carbone, la respiration mitochondriale ne peut se produire et la cellule va utiliser la glycolyse en anaérobie en coopération avec la fermentation. Même si l'ATP est consommé pour l'entretien des cellules, il ne tend pas vers zéro car les cellules peuvent survivre avec un apport sucré seulement.

<i>Environnements</i>	FA	exO2	GLC	AA	ATP (0-2)	O2 (0-1)	GLYC (0-2)	nLBP (0-1)	LBP (0-1)	FERM (0-1)	Krebs (0-2)	PHOX (0-1)	NADH (0-1)	NCD (0-2)
absence de lipides et d'oxygène														
1	0	0	0	0	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)
2	0	0	0	1		td(0)								
3	0	0	0	2		td(0)								
4	0	0	1	0	!td(0)	td(0)	osc(0-1)			!td(0)		td(0)	osc	td(0)
5	0	0	1	1	osc	td(0)	osc(0-1)	!td(0)		!td(0)		td(0)	osc	td(1)
6	0	0	1	2	osc	td(0)	osc	!td(0)	!td(0)	!td(0)		td(0)	osc	td(2)
7	0	0	2	0	!td(0)	td(0)	osc			!td(0)		td(0)	osc	td(0)
8	0	0	2	1	osc	td(0)	osc	!td(0)		!td(0)		td(0)	osc	td(1)
9	0	0	2	2	osc	td(0)	osc	!td(0)	!td(0)	!td(0)		td(0)	osc	td(2)
Pas de lipides mais apport d'oxygène														
10	0	1	0	0	td(0)	!td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)
11	0	1	0	1		!td(0)								
12	0	1	0	2		!td(0)								
13	0	1	1	0	!td(0)	!td(0)	osc			td(0)	osc	osc	osc	td(0)
14	0	1	1	1	osc	!td(0)	osc	!td(0)		td(0)	osc	osc	osc	td(1)
15	0	1	1	2	osc	!td(0)	osc	!td(0)	!td(0)	td(0)	osc	osc	osc	td(2)
16	0	1	2	0	!td(0)	!td(0)	osc			!td(0)		!td(1)	osc	td(0)
17	0	1	2	1	osc	!td(0)	osc	!td(0)	!td(0)	!td(0)		!td(1)	osc	td(1)
18	0	1	2	2	osc	!td(0)	osc	!td(0)	!td(0)	!td(0)		!td(1)	osc	td(2)
Présence de lipides mais pas d'apport en oxygène														
19	1	0	0	0	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)
20	1	0	0	1		td(0)								
21	1	0	0	2		td(0)								
22	1	0	1	0	osc	td(0)	osc(0-1)		!td(0)	!td(0)		td(0)	osc	td(0)
23	1	0	1	1	osc	td(0)	osc(0-1)	!td(0)	!td(0)	!td(0)		td(0)	osc	td(1)
24	1	0	1	2	osc	td(0)	osc	!td(0)	!td(0)	!td(0)		td(0)	osc	td(2)
25	1	0	2	0	osc	td(0)	osc		!td(0)	!td(0)		td(0)	osc	td(0)
26	1	0	2	1	osc	td(0)	osc	!td(0)	!td(0)	!td(0)		td(0)	osc	td(1)
27	1	0	2	2	osc	td(0)	osc	!td(0)	!td(0)	!td(0)		td(0)	osc	td(2)
Présence de lipides et d'oxygène														
28	1	1	0	0	td(0)	!td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)
29	1	1	0	1		!td(0)								
30	1	1	0	2		!td(0)								
31	1	1	1	0	osc	!td(0)	osc		!td(0)	td(0)	osc	osc	osc	td(0)
32	1	1	1	1	osc	!td(0)	osc	!td(0)	!td(0)	td(0)	osc	osc	osc	td(1)
33	1	1	1	2	osc	!td(0)	osc	!td(0)	!td(0)		osc	osc	osc	td(2)
34	1	1	2	0	osc	!td(0)	osc		!td(0)	!td(0)		!td(1)	osc	td(0)
35	1	1	2	1	osc	!td(0)	osc	!td(0)	!td(0)	!td(0)		!td(1)	osc	td(1)
36	1	1	2	2	osc	!td(0)	osc	!td(0)	!td(0)	!td(0)		!td(1)	osc	td(2)

Table 9.1 – Matrice de validation pour le modèle de la régulation « générale » du métabolisme cellulaire. Chaque ligne (resp. colonne) représente une condition environnementale (resp. une variable interne). De plus chaque case de la table formalise le comportement connu d'une variable dans un environnement. Les cases grises indiquent l'absence d'une connaissance ou que la connaissance ne fait pas consensus pour l'ensemble des biologistes. Les lignes 16, 17, 18, 34, 35 et 36 exhibent la glycolyse en aérobie.

NAD^+ est consommé pendant la glycolyse et régénéré lors de la fermentation (*FERM* ne tend pas vers 0); l'apport de glucose permet la production de lactate, qui réduit le $NADH$ en NAD^+ ,

qui est recyclé en *NADH* par la glycolyse, donc *NADH* oscille. En absence d'oxygène, *O2* tend vers 0 et les mitochondries ne peuvent donc pas effectuer la phosphorylation oxydative donc l'activité de *PHOX* tend vers 0. Enfin, *GLC=1* dénote un apport en glucose insuffisant pour que le flux glycolytique atteigne son taux le plus élevé, donc *GLYC* oscille entre 0 et 1.

↪ **5** – *FA=0, exO2=0, GLC=1, AA=1* : avec un apport de glucose suppléé d'acides aminés la production de biomasse non lipidique est permise, donc *nLBP* ne tend pas vers 0. Ce processus anabolique consomme *ATP*, et comme il est également produit par l'intermédiaire de la glycolyse, l'*ATP* oscille. Et, comme dans le cas de la ligne ↪4, la cellule est en anaérobie (sans oxygène) et le métabolisme cellulaire agit donc de la même manière pour les variables restantes (*NADH* et *GLYC* oscillent mais *GLYC* entre 0 et 1, *FERM* ne tend pas vers 0 et *O2* et *PHOX* tendent vers 0).

↪ **6** – *FA=0, exO2=0, GLC=1, AA=2* : un très gros apport en acides aminés favorise la phase réductrice du cycle de Krebs pour fournir les précurseurs de la synthèse des lipides, donc *LBP* ne tend pas vers 0. Les processus métaboliques sont les mêmes que ceux de la ligne ↪5, sauf qu'un apport important d'acides aminés active la glutaminolyse. Cela crée de l' α -cétoglutarate qui peut être converti par le cycle réducteur de Krebs en pyruvate. Cette accumulation de pyruvate pourrait également être due à une activité élevée de la *GLYC*, par conséquent *GLYC* pourrait parfois atteindre son niveau le plus élevé. C'est pourquoi nous préférons relaxer son comportement oscillatoire et passer de « osc(0,1) » à « osc »⁴.

↪ **7** – *FA=0, exO2=0, GLC=2, AA=0* : Correspond à un phénotype fermentaire. Avec un apport élevé de glucose, la glycolyse peut parfois atteindre son niveau le plus élevé. Donc, *GLYC* pourrait éventuellement osciller de son niveau le plus bas à son niveau le plus haut (« osc » au lieu de « osc(0,1) »). De plus, les mêmes processus que la ligne ↪ 4 sont activés : Les comportements pour les autres variables sont donc identiques à la ligne 4 (*NADH* oscille, *FERM* et *ATP* ne tendent pas vers 0 et *O2* et *PHOX* tendent vers 0).

↪ **8** – *FA=0, exO2=0, GLC=2, AA=1* : Phénotype fermentaire. Ici, l'activité catabolique (glycolyse, fermentation, respiration oxydative et Krebs) est similaire à ↪7, de sorte que *NADH* se comporte de la même manière. La production de biomasse non lipidique est possible, de sorte que *nLBP* ne tend pas vers 0. Comme c'est un processus endergonique (nécessitant de l'énergie sous forme d'*ATP*), l'*ATP* est donc produit et consommé de telle sorte qu'il oscille.

↪ **9** – *FA=0, exO2=0, GLC=2, AA=2* : ce contexte permet la production de biomasse lipidique, donc *LBP* ne tend pas vers 0. Pour les mêmes raisons que pour la ligne ↪6 les autres variables se comportent de manière similaire dans la ligne ↪8 (*nLBP* et *FERM* ne tendent pas vers 0, à l'inverse de *O2* et *PHOX*, *ATP*, *GLYC* et *NADH* qui oscillent).

9.3.2 Comportements en l'absence de lipides mais avec apport d'oxygène

Maintenant que nous avons vu l'ensemble des environnements en l'absence de lipides et d'oxygène nous traitons dans cette partie tous les phénotypes avec oxygène mais toujours sans lipides.

↪ **10** – *FA=0, exO2=1, GLC=0, AA=0* : cet environnement mène à la mort cellulaire. La cellule est maintenant en normoxie, donc *O2* ne tend pas vers 0. Tous les autres marqueurs tendent vers 0 car, sans entrées de glucose et d'acides aminés, il n'y a pas de métabolisme du glucose, donc pas de sources de carbone et de cofacteurs, pour les processus anaboliques. On s'attend donc à ce

4. L'absence de précision dans les bornes de l'oscillation indique que la formule admet les différentes oscillations possibles pour cette variables. Pour *GLYC*, qui est bornée entre 0 et 2, le pattern « osc » admet les oscillations 0-2, 1-2 et 0-1

qu'il n'y ait pas de pyruvate disponible pour le cycle de Krebs, et donc pas de phosphorylation oxydative. Enfin, comme ni la **PHOX** ni **GLYC** produisent de l'ATP, **ATP** tend vers 0.

↔ **11 & 12** – $FA=0, exO2=1, GLC=0, AA=(1 \text{ or } 2)$: as de phénotype consensus (identique aux lignes 2 & 3). Avec de l'oxygène et des acides aminés mais pas de glucose, les connaissances générales ne permettent pas de décider si l'apport carbone par les acides aminés est suffisant pour faire fonctionner la machinerie métabolique. Certaines cellules survivent (cf cellules cancéreuses lignes 2 & 3) et d'autres non. Seule la présence d'oxygène ne fait aucun doute (**O2** ne tend pas vers 0).

↔ **13** – $FA=0, exO2=1, GLC=1, AA=0$: survie en respiration aérobie. Avec un apport en sucre et en oxygène mais sans acides aminés et lipides, il y a respiration cellulaire *via* la phosphorylation oxydative. Les processus en amont de cette voie métabolique sont donc actifs et contrôlés indirectement par la quantité d'ATP : **GLYC**, **KREBS** et **PHOX** oscillent. En réponse le NADH (produit par le cycle de Krebs) oscille également. L'oxygène est présent dans la cellule et est consommé par la **PHOX**, mais en l'état des connaissances nous ne sommes pas capables d'affirmer si cette consommation est suffisante pour faire osciller l'**O2** : on ne peut donc qu'indiquer que l'oxygène ne tendra pas vers 0. L'ATP est produit mais il n'y a pas assez de connaissances générales sur sa consommation pour affirmer que **ATP** tend vers 1 ou qu'elle oscille, donc nous affirmons seulement qu'elle ne tend pas vers 0. Enfin, en fonction d'un métabolisme aérobie normal, **FERM** tend vers 0 (car la fermentation est moins efficace que la phosphorylation oxydative).

↔ **14** – $FA=0, exO2=1, GLC=1, AA=1$: condition de culture in vitro. Cette ligne peut être considérée comme représentant une cellule saine. Elle est comparable à la précédente (↔13) sauf que l'on ajoute des acides aminés. Dans ce cas tous les processus décrits précédemment ne sont pas impactés par la présence d'acides aminés. On conserve donc les mêmes comportements à une exception près : les acides aminés permettent l'activation de la voie des pentoses phosphates. Cette dernière permet la création de biomasse non lipidique (**nLBP** ne tend donc pas vers 0). De plus concernant **ATP** nous pouvons être plus précis : comme nous avons la consommation d'ATP pour la production de biomasse nous pouvons indiquer que **ATP** oscille.

↔ **15** – $FA=0, exO2=1, GLC=1, AA=2$: cette ligne est similaire à la ↔14 sauf que l'entrée importante d'acides aminés permet la production de biomasse lipidique (**LBP** ne tend pas vers 0).

↔ **16** – $FA=0, exO2=1, GLC=2 \text{ and } AA=0$: glycolyse en aérobie. **GLYC** oscille comme dans ↔14, mais la forte absorption de glucose conduit la glycolyse à un sur-fonctionnement, même en présence d'oxygène. Lorsque ce phénotype apparaît nous pouvons affirmer que la fermentation **FERM** ne tend pas vers 0 et que l'activité de la glycolyse suffit à régénérer du NADH : comme expliqué dans ↔4, **NADH** oscille. De la même manière que dans la ligne ↔13 **ATP** ne tend pas vers 0 mais on ne sait pas s'il y a consommation ou non d'ATP par les processus anaboliques (notamment production de biomasse).

De plus, étonnamment la phosphorylation oxydative pourrait être présente lorsque la glycolyse en aérobie se produit, mais probablement pas de façon constante, donc **PHOX** ne tend pas vers 1. Par conséquent, l'oxygène pourrait être partiellement consommé. Cependant nous ne pouvons pas affirmer que la consommation d'oxygène par la phosphorylation oxydative est capable de contrebalancer l'apport externe en oxygène : **O2** ne tend pas vers 0.

↔ **17 & 18** – $FA=0, exO2=1, GLC=2, AA=(1 \text{ or } 2)$: la glycolyse en aérobie se produit comme dans ↔13. De plus, l'apport en acides aminés permet la production de biomasse (permet seulement la production de biomasse non lipidique lorsque $AA=1$), de sorte que **nLBP** et **BLP** ne tendent pas vers 0 (lorsque $AA=1$ il n'y a que **nLBP** qui ne tend pas vers 0). Par conséquent l'ATP est consommé par ces processus anaboliques et donc **ATP** oscille.

9.3.3 Comportements avec un apport en lipides mais sans oxygène

Beaucoup des comportements des lignes avec apport en lipides ($FA=1$) décrivent des comportements similaires aux lignes sans apport de lipides. Pour la suite de la description, plusieurs lignes semblables sont comparées aux lignes sans lipides. Une des différences majeures dans les comportements se porte sur l'état de la variable LBP . En effet avec un apport de lipides on comprend facilement que la production de lipides complexes devient active lorsque les cofacteurs nécessaires sont présents.

\rightsquigarrow **19** – $FA=1, exO2=(0 \text{ or } 1), GLC=0, AA=0$: environnement amenant à la mort cellulaire. L'apport en lipides ne suffit pas à soutenir toutes les voies métaboliques dépendantes du carbone. L'activité de la cellule est donc, comme dans $\rightsquigarrow 1$: tous les marqueurs tendent vers 0.

\rightsquigarrow **20 & 21** – $FA=1, exO2=0, GLC=0, AA=(1 \text{ or } 2)$: aucun phénotype consensus. Comme dans les lignes $\rightsquigarrow 2 \& 3$, il n'y a pas de connaissance générale pour affirmer que l'apport lipidique seul est capable de soutenir l'activité métabolique dépendante du carbone de la cellule : on considère que le futur phénotype de la cellule est inconnu. Il est cependant possible d'affirmer que l'on aura une hypoxie cellulaire ($O2$ tend vers 0)

\rightsquigarrow **22 à 27** – $FA=1, exO2=0, GLC=(1 \text{ or } 2), AA=(0, 1 \text{ or } 2)$: ces environnements conduisent à la respiration anaérobie. En comparaison avec $\rightsquigarrow 4$ à $\rightsquigarrow 9$ l'apport d'acides gras permet, en plus de toutes les voies actives, la production de biomasse lipidique (LBP ne tend pas vers 0). Comme la production de biomasse est consommatrice d'énergie, ATP oscille. Toutes les autres variables conservent les comportement déjà décrits pour les lignes $\rightsquigarrow 4$ à $\rightsquigarrow 9$ ($\rightsquigarrow 22$ correspondant aux comportements de $\rightsquigarrow 4, \rightsquigarrow 23$ à ceux de $\rightsquigarrow 5$ etc.).

En ce point nous avons vu tous les comportements qui n'ont pas simultanément un apport en oxygène et en lipides. C'est le sujet que la prochaine section.

9.3.4 Comportements avec apport en lipides et en oxygène

\rightsquigarrow **28** – $FA=1, exO2=(0 \text{ or } 1), GLC=0, AA=0$: environnement amenant à la mort cellulaire. L'apport en lipides ne suffit pas à soutenir toutes les voies métaboliques dépendantes du carbone. L'activité de la cellule est donc comme dans $\rightsquigarrow 10$: tous les marqueurs tendent vers 0, sauf l'oxygène qui ne tend pas vers 0.

\rightsquigarrow **29 & 30** – $FA=1, exO2=1, GLC=0, AA=(1 \text{ or } 2)$: aucun consensus pour le phénotype. Comme pour $\rightsquigarrow 11 \& 12$, la connaissance générale ne nous permet pas de décider si l'apport en oxygène et en lipides est suffisant pour maintenir l'activité métabolique de la cellule dépendante au carbone (cases grises). Seule la normoxie ne fait aucun doute ($O2$ ne tend pas vers 0).

\rightsquigarrow **31** – $FA=1, exO2=1, GLC=1, AA=0$: survie en respiration en aérobie. En comparaison avec la ligne $\rightsquigarrow 13$, seule la présence de lipides change. Dans ce cas cela indique qu'en présence d'ATP la production de biomasse lipidique est possible ($nLBP$ ne tend pas vers 0). De plus la consommation d'ATP par la production de biomasse indique que la variable ATP oscille. Pour tous les autres marqueurs la cellule exhibe exactement les mêmes comportement que pour la ligne $\rightsquigarrow 13$.

\rightsquigarrow **32** – $FA=1, exO2=1, GLC=1, AA=1$: par rapport à $\rightsquigarrow 14$ qui correspond à des conditions optimales de culture cellulaire, l'apport en acides gras ne fait que soutenir la production de biomasse lipidique, donc LBP ne tend pas vers 0 et les autres marqueurs conservent les mêmes comportements que pour la ligne $\rightsquigarrow 14$.

↪ **33** – $FA=1, exO2=1, GLC=1, AA=2$: respiration en aérobie. En comparaison avec ↪ 32, la glutaminolyse alimente Krebs (réactions anapérotyques) qui permet la création de citrate et son accumulation. A partir de citrate il est possible de créer des lipides *via* le multiplexe **LS**. Cet excès de citrate rend éventuellement possible l'alimentation de la fermentation en plus de la production de lipides (qui est par ailleurs déjà possible à l'aide de l'apport en lipides). Ainsi, par précaution, nous n'indiquons pas de comportement pour **FERM**.

↪ **34 to 36** – $FA=1, exO2=1, GLC=0, AA=(0,1 \text{ or } 2)$: glycolyse en aérobie. Comparé aux lignes de ↪16 à ↪18, nous avons un apport supplémentaire de lipides. cet apport permet de produire des lipides complexes : **LBP** ne tend pas vers 0. De plus la production biomasse lipidique nécessite comme déjà indiqué précédemment la consommation d'ATP : **ATP** oscille. Pour toutes les autres variables l'effet Warburg explique les comportements, et ce, en accord avec les descriptions faites respectivement pour les lignes ↪16 à ↪18.

La matrice de validation, Table 9.1, proposée dans cette section sera ensuite utilisée comme input de *GreenBioNet* que nous avons développé à cette fin au sein de *TotemBioNet*. Après avoir terminé la partie formalisation des phénotypes, il ne nous reste plus qu'à passer par l'étape d'identification des paramètres (voir Chapitre 5). L'identification des paramètres s'est faite dans un premier temps de manière manuelle à l'aide des connaissances biochimiques locales sur les régulations des différents composants du métabolisme. Après l'étape d'identification manuelle, certains paramètres resteront non identifiés. L'ensemble des paramètres non identifiés peuvent prendre toutes les valeurs possibles. C'est à cette étape que nous passerons à l'utilisation de *TotemBioNet*. Cet outil nous permettra d'énumérer l'ensemble des paramétrisations possibles en traitant environnement par environnement la vérification de la cohérence entre la matrice proposée dans la précédente section et le réseau de régulation proposé.

9.4 Spécifications détaillées : identification manuelle des paramètres

En raison du niveau d'abstraction à gros grains de notre modèle, où les variables et les multiplexes font principalement référence à des marqueurs biochimiques classiques, nous profitons d'un grand nombre de connaissances scientifiques qui ont été accumulées et recensées dans différentes références bibliographiques [Nelson and Cox, 2012, Koolman and Röhm, 2012, Berg et al., 2002] pour l'identification manuelle des paramètres. Cette dernière se base sur l'expérience de pensée présentée dans le chapitre 5 du manuscrit. L'ensemble des paramètres dont la valeur a été identifiée est présenté dans le tableau 9.2. Enfin, parmi l'ensemble des paramètres seuls 7 n'ont pas pu être identifiés faute de connaissances biochimiques. Pour l'ensemble de la section, les paramètres sont décrits en fonction des variables auxquelles ils sont associés.

9.4.1 Paramètres des variables représentant des voies métaboliques

9.4.1.1 Les paramètres de *GLYC* (Figure 9.4)

Pour mener à bien chaque expérience de pensée et identifier au mieux les paramètres pour chaque variable il faut avoir en tête toutes les ressources possibles et toutes les cibles de la variable. La Figure 9.4 se concentre sur ces informations pour la variable **GLYC**. Cette variable active potentiellement dès son niveau 1 toutes ses cibles, mais, en fonction des conditions extérieures, le niveau 2 peut être requis. Par exemple le niveau 2 de **GLYC** est nécessaire pour agir sur trois cibles :

<p># Paramètres de l'ATP $K_{ATP} = 0$ $K_{ATP, LBP} = 0$ $K_{ATP, nLBP} = 0$ $K_{ATP, LBP nLBP} = 0$ $K_{ATP, PHOX} = 1$ $K_{ATP, nLBP PHOX} = 1$ $K_{ATP, LBP nLBP PHOX} = 2$ $K_{ATP, GLYC1} = 0$ $K_{ATP, GLYC1 nLBP} = 0$ $K_{ATP, GLYC1 LBP nLBP} = 2$ $K_{ATP, GLYC1 GLYC2} = 1$ $K_{ATP, GLYC1 GLYC2 nLBP} = 1$ $K_{ATP, GLYC1 GLYC2 LBP nLBP} = 2$ $K_{ATP, GLYC1 PHOX} = 1$ $K_{ATP, GLYC1 nLBP PHOX} = 1$ $K_{ATP, GLYC1 LBP nLBP PHOX} = 2$ $K_{ATP, GLYC1 GLYC2 PHOX} = 1$ $K_{ATP, GLYC1 GLYC2 nLBP PHOX} = 1$ $K_{ATP, GLYC1 GLYC2 LBP nLBP PHOX} = 2$</p>	<p># Paramètres de l'O2 $K_{O2} = 0$ $K_{O2, PHOX} = 0$ $K_{O2, exO2} = 1$ $K_{O2, exO2 PHOX} = 1$</p> <p># Paramètres de GLYC $K_{GLYC} = 0$ $K_{GLYC, GR} = 0$ $K_{GLYC, GLC1} = 0$ $K_{GLYC, GLC1 GR} = 0$ $K_{GLYC, GLC1 GLC2} = 0$ $K_{GLYC, GLC1 GLC2 GR} = 0$ $K_{GLYC, COF} = 0$ $K_{GLYC, COF GR} = 0$ $K_{GLYC, COF GLC1} = 0$ $K_{GLYC, COF GLC1 GR} = 1$ $K_{GLYC, COF GLC1 GLC2} = 1$ $K_{GLYC, COF GLC1 GLC2 GR} = 2$</p> <p># Paramètres de nLBP $K_{nLBP} = 0$ $K_{nLBP, PPP} = 1$ $K_{nLBP, AAS} = 1$ $K_{nLBP, AAS PPP} = 1$</p> <p># Paramètres de Krebs $K_{Krebs} = 0$ $K_{Krebs, AnO} = 1$ $K_{Krebs, AnO a-KG} = 2$ $K_{Krebs, BOX} = 1$</p> <p># Paramètres de PHOX $K_{PHOX} = 0$ $K_{PHOX, PC} = 1$</p>	<p># paramètres de NADH $K_{NADH} = 0$ $K_{NADH, FERM} = 0$ $K_{NADH, PHOX} = 0$ $K_{NADH, AAS} = 0$ $K_{NADH, FERM PHOX} = 0$ $K_{NADH, FERMAAS} = 0$ $K_{NADH, AAS PHOX} = 0$ $K_{NADH, FERMAAS PHOX} = 0$ $K_{NADH, FERM Krebs AAS} = 0$ $K_{NADH, GLYC AAS PHOX} = 0$ $K_{NADH, GLYC FERMAAS PHOX} = 1$ $K_{NADH, Krebs FERMAAS PHOX} = 1$ $K_{NADH, GLYC Krebs FERMAAS PHOX} = 1$ $K_{NADH, GLYC Krebs AAS} = 0$ $K_{NADH, FERM GLYC AAS} = 0$ $K_{NADH, GLYC Krebs} = 0$ $K_{NADH, FERM Krebs} = 0$ $K_{NADH, Krebs AAS} = 0$ $K_{NADH, GLYC AAS} = 0$ $K_{NADH, GLYC PHOX} = 0$ $K_{NADH, FERM GLYC} = 0$ $K_{NADH, Krebs} = 0$ $K_{NADH, GLYC} = 0$ $K_{NADH, FERM GLYC Krebs AAS} = 1$ $K_{NADH, Krebs PHOX} = 1$ $K_{NADH, Krebs FERM PHOX} = 1$ $K_{NADH, Krebs AAS PHOX} = 1$ $K_{NADH, GLYC Krebs PHOX} = 1$ $K_{NADH, FERM GLYC Krebs PHOX} = 1$ $K_{NADH, GLYC Krebs AAS PHOX} = 1$</p> <p># Paramètres de FERM $K_{FERM} = 0$ $K_{FERM, EP} = 1$</p>
--	--	--

Table 9.2 – Valeurs des paramètres de la section 9.4. Seulement 7 valeurs des paramètres restent libres : $K_{ATP, LBP PHOX}$, $K_{ATP, GLYC1 LBP}$, $K_{ATP, GLYC1 GLYC2 LBP}$, $K_{ATP, GLYC1 LBP PHOX}$, $K_{ATP, GLYC1 GLYC2 LBP PHOX}$, $K_{NADH, FERM GLYC PHOX}$ et $K_{NADH, FERM GLYC Krebs}$.

EP si $O2 = 1$ et $NADH \geq 1$,

α **KG** si $NCD < 2$ et $O2 = 1$,

sur-activation de l'**ATP** (en comparaison avec une activation « simple » du niveau 1).

1. Dans la première phase de raisonnement pour l'expérience de pensée, nous devons mettre de côté les paramètres non opérables : ce sont ceux qui ont des *ressources non consistantes* vis à vis du graphe de régulation (Figure 9.3). Par exemple ici **GLC2** comme ressource implique forcément que **GLC1** soit présent lui aussi comme ressource (car tous deux représentent des niveaux de la même variable). C'est pourquoi $K_{GLYC, GLC2}$, $K_{GLYC, COF GLC2}$, $K_{GLYC, COF GLC2 GR}$ et $K_{GLYC, GLYC2 GR}$ sont des paramètres non opérables. Il n'ont donc pas besoin d'être instanciés car ils ne seront jamais utilisés pour définir la dynamique du réseau de régulation. Sachant que **GLYC** a 4 prédécesseurs et que 4 de ses paramètres sont non opérables, cela signifie qu'il reste $2^4 - 4 = 12$ paramètres à estimer pour **GLYC**.
2. La seconde phase du raisonnement traite les cas « faciles », principalement quand la variable traitée est attirée en 0 par *manque de ressources*. Dans le cas de **GLYC**, les ressources **GLC** et **COF** (cofacteurs) sont nécessaires pour le bon fonctionnement de la gly-

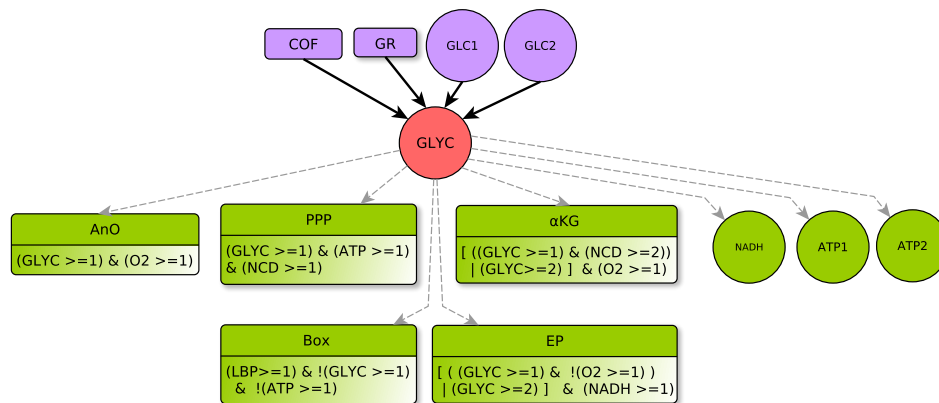


Figure 9.4 – Ressources et cibles de la variable représentant la glycolyse

colyse. Si au moins un des deux manque à l'appel, la glycolyse ne peut pas s'effectuer convenablement et elle devient inactive sur ses cibles. Cela implique que 8 paramètres qui n'ont pas à la fois **COF** et **GLC** comme ressource sur les 12 à identifier sont égaux à 0 K_{GLYC} , $K_{GLYC, GLC1}$, $K_{GLYC, COF}$, $K_{GLYC, GR}$, $K_{GLYC, GLC1 GLC2}$, $K_{GLYC, GLC1 GR}$, $K_{GLYC, COF GR}$ et $K_{GLYC, GLC1 GLC2 GR}$.

- D'autres cas faciles se retrouvent quand la variable est sous l'emprise de toutes ces ressources. Cela signifie que tous les activateurs sont présents et qu'aucun inhibiteur n'est présent. Le multiplexe **GR** représente une inhibition de **KREBS** et **ATP** (par l'inhibition de la Phosphofructokinase). Quand **GR** est une ressource de **GLYC**, si les cofacteurs et le glucose sont présents, et si le niveau de glucose est au plus haut, la glycolyse peut atteindre son plus haut niveau :
 - quand **GLC** = 2 le fonctionnement de la glycolyse est tel que l'on peut activer le multiplexe **EP** même si **O2** = 1 et $NADH \geq 1$, comme le multiplexe **αKG** si $NCD < 2$ et **O2** = 1. Pour ces raisons $K_{GLYC, COF GLC1 GLC2 GR} = 2$.
 - quand **GLC** = 1 le fonctionnement de la glycolyse permettra d'activer toutes ses cibles (si toutes les autres conditions sont satisfaites), mais pas le multiplexe **EP** si **O2** = 1 et $NADH \geq 1$, et non plus le multiplexe **αKG** si $NCD < 2$ et **O2** = 1. C'est pourquoi $K_{GLYC, COF GLC1 GR} = 1$.
- La dernière phase de raisonnement lors d'une expérience de pensée consiste à identifier les paramètres avec un ensemble de ressources intermédiaires. Ici, deux paramètres ont encore besoin d'être identifiés : $K_{GLYC, COF GLC1 GLC2}$ et $K_{GLYC, COF GLC1}$. Dans les deux cas glucose, (**GLC**), et cofacteurs, (**COF**) sont présents et l'inhibiteur **GR** est présent réduisant donc l'efficacité de la glycolyse. D'après les connaissances biologiques, l'inhibition *via* **GR**, même en présence d'une très grande quantité de glucose stoppe la « sur-activation » de **ATP** et **EP** quand **O2** = 1 et $NADH \geq 1$. C'est pourquoi $K_{GLYC, COF GLC1 GLC2} = 1$. Avec un apport normal en glucose, **GR** est assez puissant pour arrêter le processus glycolytique et empêcher la glycolyse d'activer ses cibles ; c'est pourquoi $K_{GLYC, COF GLC1} = 0$.

Dans la suite de cette section, l'utilisation des 4 phases de l'expérience de pensée est aussi effectuée. Cependant dans certains cas, l'abstraction de la connaissance biologique peut être sujette à interprétations et nous prenons soin de rendre dans tous nos choix l'interprétation explicite. Ces

choix peuvent être considérés comme des hypothèses raisonnables dans le cadre de notre étude sur la régulation du métabolisme des eucaryotes génériques. A la fin du processus de modélisation et pour répondre à une question différente ou lorsque les connaissances biologiques le permettront ces interprétations pourront éventuellement être réévaluées.

9.4.1.2 Les paramètres de *KREBS* (Figure 9.5)

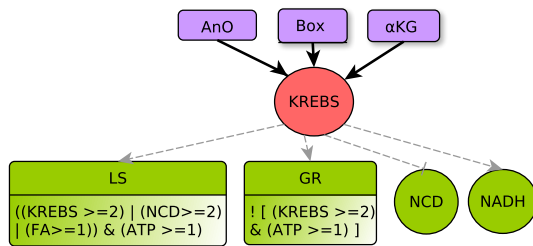


Figure 9.5 – les ressources et les cibles la variable *KREBS*.

KREBS joue un rôle double en fonction de l’environnement de la cellule. Au niveau 1, *KREBS* est un producteur de *NADH*. Au niveau 2, *KREBS* est en sur-fonctionnement et donne un signal d’alarme pour réduire l’activation de la glycolyse (*GR* est le multiplexe de régulation de la glycolyse) et ainsi diminuer le niveau de pyruvate sortant de la glycolyse. Toujours au niveau 2, *KREBS* provoque également la production de lipides (multiplexe *LS*) et consomme des composés issus de la dégradation des acides aminés *NCD*. *LS*.

Les ressources de *KREBS* sont : *BOX* via la dégradation des acides gras, α *KG* provenant de la glutaminolyse, et *AnO* car l’acétyl-CoA est dérivé du pyruvate, le produit final de la glycolyse. Ainsi, il y a $2^3 = 8$ paramètres à identifier pour *KREBS* car il a 3 prédécesseurs.

1. *Les paramètres non opérables* : en analysant les formules des multiplexes *AnO* et α *KG* respectivement $[(GLYC \geq 1) \& (O2 \geq 1)]$ et $[(GLYC \geq 1) \& (NCD \geq 2) \mid (GLYC \geq 2)] \& (O2 \geq 1)$; on constate que la formule de α *KG* correspond à une sous partie de la formule d’*AnO*. Cela sous entend que lorsque le multiplexe *AnO* est ressource, alors le multiplexe α -*KG* aussi, ceci mettant en exergue que les deux multiplexes régulent par l’intermédiaire de la glycolyse. De plus, la satisfaction de la formule $[(LBP \geq 1) \& !(GLYC \geq 1) \& !(ATP \geq 1)]$ du multiplexe *BOX* implique que *GLYC*=0, et contredit donc les formules des deux autres multiplexes précédemment cités : α *KG* and *AnO*. Au final les 4 paramètres : $K_{Krebs, \alpha KG}$, $K_{Krebs, BOX \alpha KG}$, $K_{Krebs, AnO \alpha KG}$ and $K_{Krebs, AnO BOX}$ sont donc non opérables.
2. *Absence d’activateurs* : quand aucune des trois ressources de *KREBS* n’est disponible, la variable n’agit sur aucune de ses cibles : $K_{Krebs} = 0$.
3. *Ensemble de toutes les ressources* : Lorsque *KREBS* a toutes ses ressources (α *KG* et *AnO*), une alarme est donnée via l’activation pour inhiber les processus cataboliques tels que la glycolyse (via *GR*) ou la dégradation acides aminés (*NCD*). Tout ceci évite la surproduction d’énergie. L’alarme donnée par *KREBS* favorise également les processus anaboliques (activation de *LS*). Ainsi, $K_{Krebs, AnO \alpha KG} = 2$.
4. *Ensemble intermédiaire de ressources* : Avec un apport normal d’acétyl-coA par la glycolyse (*AnO*), *KREBS* produit du *NADH* et ce, sans alarmer la cellule. Ainsi $K_{Krebs, AnO} = 1$, et pour la même raison $K_{Krebs, BOX} = 1$.

9.4.1.3 Les paramètres de *FERM*(Figure 9.6)

Les paramètres de la variable *FERM* sont des cas faciles parce qu'ils ont seulement deux valeurs possibles (*FERM* est booléenne) et *FERM* a une seule ressource le multiplexe **EP** (qui dénote un excès de pyruvate). Il y a donc 2 paramètres seulement qui correspondent respectivement aux cas sans aucune ressource et au cas avec toutes les ressources. Si la ressource est absente, alors la fermentation n'est pas capable de produire du NADH, donc $K_{FERM} = 0$. Inversement, si la ressource est présente, il produit du NADH, donc $K_{FERM, EP} = 1$.

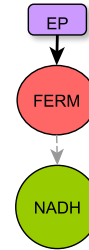


Figure 9.6 – Ressources et cibles de *FERM*.

9.4.1.4 Les paramètres de *PHOX*(Figure 9.7)

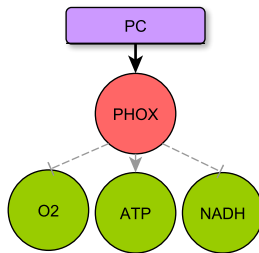


Figure 9.7 – Ressources et cibles de *PHOX*

Les paramètres de la variable *PHOX* sont aussi des cas faciles car *PHOX* possède les mêmes caractéristiques que *FERM*. C'est une variable booléenne et elle n'a qu'une seule ressource possible. On a donc suivi le même schéma qu'avec *FERM* pour identifier les paramètres de *PHOX*. *PHOX* est capable de consommer de l'oxygène et du NADH et produire de l'ATP que lorsque sa ressource **PC** (comme son nom l'indique **PC** indique « Phox-control ») est disponible donc $K_{FERM} = 0$ et $K_{PHOX, PC} = 1$

9.4.2 Paramètres des variables représentant la biomasse

9.4.2.1 Les paramètres de *nLBP*(Figure 9.8)

La variable *nLBP* (production de biomasse non lipidique) est booléenne, agit négativement sur sa cible *ATP* lorsque *nLBP*=1 et a 2 ressources possibles : **PPP** et **AAS**, dont les combinaisons sont toutes satisfiables, ce qui signifie que *nLBP* a 4 paramètres identifiables.

1. Pas de paramètres non applicable
2. *Absence d'activateurs* : quand ni la synthèse de nucléotides (**PPP**) ni la synthèse d'acides aminés (**AAS**) ne sont ressources, la production de biomasse non lipidique est stoppée et ne consomme pas d'ATP : $K_{nLBP} = 0$.
3. *Ensemble de toutes les ressources* : quand **PPP** et **AAS** sont présents alors il y a production de biomasse et consommation d'ATP : $K_{nLBP, AAS PPP} = 1$.
4. *Ensemble intermédiaire de ressources* : les processus **PPP** comme **AAS** ont besoin d'ATP pour fonctionner. La présence d'une des deux ressources de *nLBP* doit donc être suffisante pour inhiber la variable *ATP* donc $K_{nLBP, PPP} = K_{nLBP, AAS} = 1$.

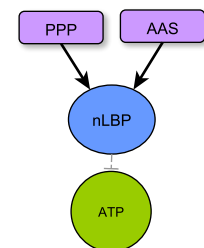


Figure 9.8 – Ressources et cibles de *nLBP*.

9.4.2.2 Les Paramètres de *LBP* (Figure 9.9)

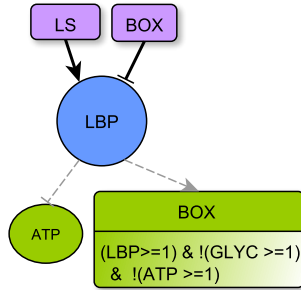


Figure 9.9 – Ressources et cibles de *LBP*.

La variable concernant la production de lipides complexes, *LBP*, est également booléenne et de manière concomitante consomme *ATP* et favorise *BOX* (β -oxydation ou dégradation des Acides gras). Les ressources possibles de *LBP* sont les multiplexes *LS* (synthèse de lipides) et *BOX*, ce qui fait que l'expérience de pensée associée pour l'identification de ces paramètres est particulièrement subtile. En effet même si *BOX* est à la fois cible et ressource de *LBP* formant une boucle de rétroaction, il faut garder à l'esprit que d'un point de vue formel cibles et ressources sont indépendantes. En d'autres termes, on doit considérer que *BOX* en tant que ressource est différente de *BOX* comme cible.

1. Il n'y a pas de paramètres non applicable pour *LBP* car il n'y a pas d'incompatibilité dans les formules des multiplexes ressources de la variable. Il y a donc 4 paramètres à identifier.
2. *Absence d'activateurs* : Sans ressources, pas d'activateur (*LS*) et présence de l'inhibiteur (*BOX*), la variable *LBP* n'est pas capable d'activer ses cibles : $K_{LBP} = 0$. De plus, même en enlevant l'inhibiteur, sans activateur la production de biomasse lipidique n'est pas suffisante pour consommer de l'*ATP* ni activer la voie de dégradation des lipides (*BOX*) donc $K_{LBP, BOX} = 0$.
3. *Ensemble de toutes les ressources* : quand la synthèse des lipides est effective sans présence de l'inhibiteur, la production et stockage de biomasse lipidique provoquent la consommation d'*ATP* et l'activation de la voie de dégradation des lipides donc $K_{LBP, LS, BOX} = 1$.
4. *Ensemble intermédiaire de ressources* : il reste le cas où l'on a l'activateur et l'inhibiteur présents ; dans ce cas la consommation d'*ATP* par la synthèse de lipides est toujours effective, *LBP* doit donc activer la consommation d'*ATP* et $K_{LBP, LS} = 1$.

9.4.3 Paramètres des variables représentant des métabolites

9.4.3.1 Les paramètres de *NCD* (Figure 9.10)

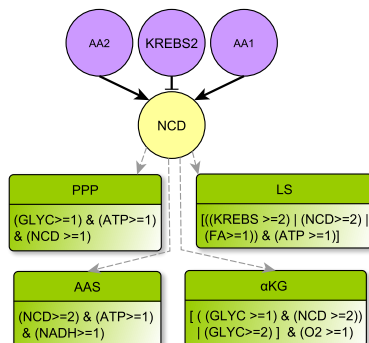


Figure 9.10 – Ressources et cibles de *NCD*.

NCD est un fournisseur d'azote et de carbone provenant des acides aminés. Dans notre modèle, *AA* fournit en acides aminés *NCD* à deux niveaux différents et *KREBS* en est un inhibiteur. Il y a donc $2^3 = 8$ paramètres. Au niveau 1, *NCD* permet la création de nucléotides et d'ADN en fournissant de l'azote via *PPP*. Au niveau 2, il permet de créer α *KG* par la glutaminolyse, des acides aminés via *AAS*, et des lipides via *LS* en fournissant de l'azote et du carbone.

1. *Paramètres non opérables* : comme l'apport en acides aminés se fait par l'intermédiaire de deux niveaux de la variable **AA**, avoir comme ressource **AA** au niveau 2 implique forcément que **AA** au niveau 1 est ressource. Ceci rend non opérables les paramètres $K_{NCD,AA2}$ et $K_{NCD,AA2\text{Krebs}}$ qui contiennent **AA** au niveau 2 sans **AA** au niveau 1. Il ne reste donc plus que 6 paramètres à identifier.
2. *Absence d'activateurs* : Sans aucune ressource le paramètre K_{NCD} est sans activateur (sans apport suffisant d'acides aminés) et en présence de son inhibiteur, il n'est pas capable d'activer ses cibles donc $K_{NCD} = 0$. Sans inhibiteur, mais toujours sans activateurs on considère que $K_{NCD,\text{Krebs}} = 0$ car **NCD** n'active aucune de ses cibles.
3. *Ensemble de toutes les ressources* : lorsque **NCD** est en présence de toutes ses ressources, aucun inhibiteur n'est présent et tous les activateurs le sont : il y a entrée d'acides aminés dans la cellule mais aucune utilisation des composés carbonés et azotés de ces mêmes acides aminés par le cycle de Krebs. La variable est donc capable d'activer l'ensemble de ses cibles et $K_{NCD,AA1\text{AA2}\text{Krebs}} = 2$.
4. *Ensemble intermédiaire de ressources* : L'inhibiteur **KREBS** est capable de réduire la vitesse de production des composés carbonés et azotés issus des acides aminés. Cependant comme l'expérience de pensée doit considérer le comportement asymptotique cela ne réduit pas la valeur vers laquelle vont tendre les niveaux d'azote et de carbone disponibles pour l'utilisation dans différentes voies métaboliques car techniquement de manière locale il n'y a pas de dégradation de ces composés. Cela indique que $K_{NCD,AA1\text{AA2}} = K_{NCD,AA1\text{AA2}\text{Krebs}} = 2$ et que $K_{NCD,AA1} = K_{NCD,AA1\text{Krebs}}$.
Enfin, comme $K_{NCD,AA1} = K_{NCD,AA1\text{Krebs}}$ il reste à définir la valeur de $K_{NCD,AA1\text{Krebs}}$ pour finir l'identification de l'ensemble des paramètres de **NCD**. Dans ce cas **AA**=1 et il y a suffisamment d'apport d'azote pour synthétiser des nucléotides *via* la **PPP** mais pas assez de production d'azote et de carbone pour être actif sur synthèse d'acides aminés **AAS** ou de lipides **LS**, ainsi que **α KG**, donc $K_{NCD,AA1\text{Krebs}} = 1$ (et $K_{NCD,AA1} = 1$).

9.4.3.2 Les paramètres de **O2** (Figure 9.11)

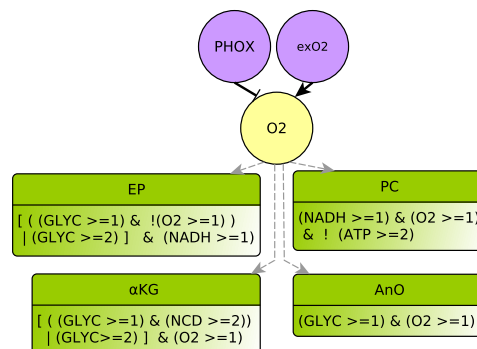


Figure 9.11 – Ressources et cibles de **O2**.

L'activateur de l'**O2** est l'oxygène extérieur (**exO2**) et l'inhibiteur de **O2**, la phosphorylation oxydative (**PHOX**) qui consomme de l'oxygène. Ils sont indépendants l'un de l'autre ; les 4 paramètres de **O2** sont donc opérables. Toutes les cibles de **O2**, à savoir **AnO**, **αKG**, **EP** et **PC**, partagent le même seuil : **O2** est simplement présent ou absent.

1. *Absence d'activateurs* : Bien entendu sans ou avec peu d'apport d'oxygène de l'environnement cellulaire, quelle que soit la présence ou l'absence de l'inhibiteur, il n'y aura pas/peu d'oxygène dans la cellule⁵ par conséquent **O2** n'activera aucune de ses cibles et $K_{O_2} = K_{O_2, PHOX} = 0$.
2. *Ensemble de toutes les ressources* : lorsqu'il y a un apport d'oxygène extérieur à la cellule via la variable **exO2** et aucun inhibiteur, il y a suffisamment d'oxygène pour que la variable **O2** active toutes ses cibles donc $K_{O_2, exO2 PHOX} = 1$.
3. *Ensemble intermédiaire de ressources* : il ne reste maintenant plus qu'un paramètre à évaluer pour cette variable. Lorsque l'on a un apport d'oxygène extérieur même si la phosphorylation oxydative fonctionne la cellule ne tombe pas en hypoxie et est capable d'utiliser l'oxygène pour réguler, en coopération avec d'autres variables, l'activation ou l'inhibition de différentes voies métaboliques. En d'autres termes **O2** est capable d'activer toutes ses cibles et $K_{O_2, exO2} = 1$.

Nous arrivons maintenant à l'identification des paramètres associés à la variable **NADH**. C'est un très gros carrefour de la régulation du métabolisme avec un grand nombre de ressources.

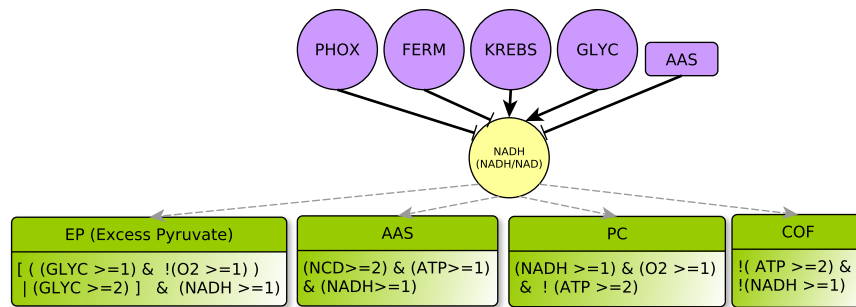
9.4.3.3 Les paramètres de **NADH** (Figure 9.12)

Rappelons que **NADH** représente le quotient $NAD(P)H / NAD(P)^+$, donc **NADH** = 1 signifie indifféremment la présence de $NAD(P)H$ ou l'absence de $NAD(P)^+$, et **NADH**=0 signifie aussi bien la présence de $NAD(P)^+$ que le manque de $NAD(P)H$. **NADH** agit comme un activateur de la phosphorylation (**PHOX** via **PC** lorsque **O2** est présent), ou comme activateur de la fermentation (**FERM** via **EP** en cas d'hypoxie ou lorsque la glycolyse produit un excès de pyruvate). Il est aussi utile à la synthèse des acides aminés (lorsque **ATP** et **NCD** sont présents). Enfin, il agit également comme inhibiteur de la glycolyse (**GLYC** via **COF**). Comme **NADH** est booléen toutes ses régulations s'appliquent au même seuil.

NADH a 2 fournisseurs qui permettent d'obtenir du **NADH** et/ou **NADPH** : **KREBS** et **GLYC** et 3 consommateurs qui augmentent la concentration en NAD^+ et/ou $NADP^+$: **PHOX**, **FERM** et **AAS**. Toutes les combinaisons de ces 5 ressources potentielles sont satisfiables, ce qui conduit à 32 paramètres à identifier.

1. *Absence d'activateurs* : Lorsque ni **KREBS** ni **GLYC** ne sont présentes. **NADH** ne va évidemment pas activer ses cibles, et ce quel que soit le nombre d'inhibiteurs présents donc $K_{NADH} = K_{NADH, FERM} = K_{NADH, PHOX} = K_{NADH, AAS} = K_{NADH, FERMPHOX} = K_{NADH, FERMAAS} = K_{NADH, AAS PHOX} = K_{NADH, FERMAAS PHOX} = 0$.
2. *Ensemble de toutes les ressources* Après avoir identifié tous les paramètres sans activateur il reste 24 paramètres à définir. Lorsque toutes les ressources sont présentes il n'y a pas

5. La cellule considérée n'étant pas une cellule végétale il n'y a pas de production d'oxygène via la photosynthèse

Figure 9.12 – Ressources et cibles de $NADH$.

d'inhibiteur et tous les activateurs, $NADH$ est capable de participer à l'activation de l'ensemble de ses cibles donc $K_{NADH, GLYC \text{ krebs FERMAAS PHOX}} = 1$.

3. *Sans inhibiteurs* : Comme $NADH$ booléen, on peut généraliser un cas particulier : lorsque aucun des inhibiteurs n'est présent mais qu'au moins un des activateurs est présent, $NADH$ est capable de participer à l'activation toutes ses cibles $K_{NADH, GLYC \text{ FERMAAS PHOX}} = K_{NADH, \text{krebs FERMAAS PHOX}} = 1$.
4. *Ensemble intermédiaire de ressources* : A ce stade il reste 21 valeurs de paramètres à identifier pour $NADH$, et nous pouvons fixer une majorité d'entre eux grâce à deux des hypothèses qui semblent judicieuses dans notre contexte :
 - (a) Dans la cellule, le $PHOX$ est le principal consommateur de $NADH$, $KREBS$ est son principal producteur, et les processus métaboliques qu'ils abstraient sont imbriqués, ils s'équilibrent sur le long terme.
 - (b) $GLYC$ est un faible producteur de $NADH$, et $FERM$ et $GLYC$ des consommateurs respectivement moyen et faible (mais nous ne savons pas si $GLYC$ et AAS s'équilibrent sur le long terme).

La première hypothèse, une fois traduite en termes de ressources, implique que $K_{NADH, \text{FERM} \text{ krebs AAS}} = 0$ comme $KREBS$ et $PHOX$ s'équilibrent, la situation est donc similaire à celle d'une absence d'activateurs. De plus, avec la neutralisation de l'inhibition de $PHOX$ par $KREBS$, plus aucun inhibiteur n'est présent (car $FREM$ et AAS sont ressources). A l'aide de la seconde hypothèse on peut identifier $K_{NADH, GLYC \text{ AAS PHOX}} = 0$ car pour ce paramètre le seul producteur $GLYC$ est faible et le consommateur moyen $FERM$ est présent dans ce cas $NADH$ n'est pas capable d'activer ses cibles. En prenant en compte l'équilibre entre $KREBS$ et $PHOX$, la deuxième hypothèse implique également $K_{NADH, GLYC \text{ krebs AAS}} = 0$ pour la même raison. On peut aussi déduire que $K_{NADH, \text{FERM} \text{ GLYC AAS}} = 0$, car $GLYC$ est ici le seul activateur et la force de l'inhibiteur principal $PHOX$ est plus importante. A l'aide de ces hypothèses nous avons mis 4 paramètres à 0, et par le principe de Snoussi explicité dans le chapitre 3, nous pouvons déduire 8 valeurs de paramètres supplémentaires (qui ont moins de ressources que les 4 paramètres que l'on vient d'identifier) : $K_{NADH, GLYC \text{ krebs}} = K_{NADH, \text{FERM} \text{ krebs}} = K_{NADH, \text{krebs AAS}} = K_{NADH, GLYC \text{ AAS}} = K_{NADH, GLYC \text{ PHOX}} = K_{NADH, \text{FERM} \text{ GLYC}} = K_{NADH, \text{krebs}} = K_{NADH, GLYC} = 0$.

A l'inverse lorsque l'équilibre est en faveur des activateurs, *NADH* sera capable d'activer ses cibles. C'est le cas pour $K_{NADH, FERMGLYCKrebs AAS} = 1$ parce que *PHOX* et *KREBS* s'équilibrent, et qu'il n'y a aucun autre inhibiteur, en conséquence l'activation par la variable *GLYC* fait la différence. Lorsque l'activation *KREBS*, la présence de *AAS* et de *FERM* n'est pas suffisante pour équilibrer l'activation car ce sont tout au plus des inhibiteurs moyens. Avec ces affirmations $K_{NADH, Krebs PHOX} = 1$. Toujours en suivant les conditions de Snoussi, tous les paramètres qui possèdent plus de ressources que le dernier identifié doivent avoir une valeur identique ou supérieure. Or, *NADH* est booléen donc on en déduit que $K_{NADH, Krebs FERM PHOX} = K_{NADH, Krebs AAS PHOX} = K_{NADH, GLYCKrebs PHOX} = K_{NADH, FERMGLYCKrebs PHOX} = K_{NADH, GLYCKrebs AAS PHOX} = 1$

En comptabilisant tous les paramètres déduits des hypothèses et des conditions de Snoussi il reste encore 2 paramètres à identifier : $K_{NADH, FERMGLYCKrebs}$ et $K_{NADH, FERMGLYCKrebs PHOX}$. Pour ces paramètres, l'unique producteur est *GLYC*. Le consommateur pour le premier paramètre est *AAS* (en considérant l'équilibre *KREBS/PHOX* donné par la première hypothèse) et pour le deuxième on a *GLYC* en plus de *AAS*. Tous deux sont des inhibiteurs faibles et nous avons décidé de ne pas ajouter d'hypothèse car elle aurait été trop arbitraire par rapport au contexte général évoqué. Nous laissons donc ces deux paramètres non identifiés.

Nous arrivons maintenant à la description des paramètres de la dixième et dernière variable l'*ATP*. L'*ATP* est central au niveau du métabolisme car il donne une indication du niveau énergétique de la cellule. Bien identifier ces paramètres est donc primordial pour élaborer un modèle énergétique de la cellule de manière générale.

9.4.3.4 Les paramètres de *ATP* (Figure 9.13)

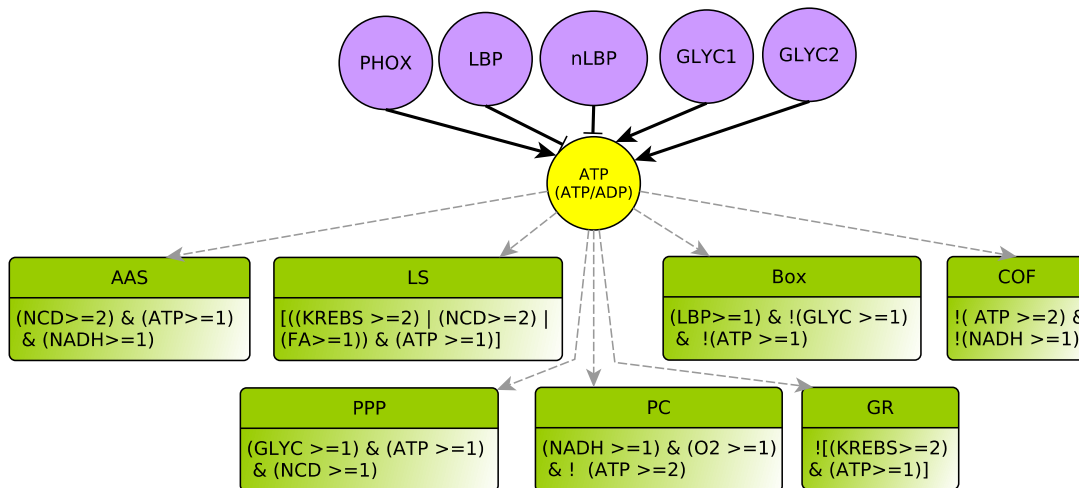


Figure 9.13 – Ressources et cibles de *ATP*.

ATP représente le quotient ATP/ADP , avec deux seuils d'activité. Au niveau 1 : il participe à la synthèse des acides aminés, des nucléotides et des lipides (**AAS**, **PPP** et **LS**), il inhibe la β -oxydation **BOX** et il régule la glycolyse quand $KREBS=2$ via **GR**. Au niveau 2, il inhibe en plus la phosphorylation oxydative par l'intermédiaire du multiplexe **PC** et régule à nouveau la glycolyse par l'intermédiaire du **COF**.

ATP a 3 producteurs : deux niveaux de glycolyse (**GLYC1** et **GLYC2**) ainsi que la phosphorylation oxydative (**PHOX**). Il a 2 consommateurs : **LBP** et **nLBP** (production de biomasse). Ce qui monte le nombre total de paramètres à $2^5 = 32$

1. *Paramètres non opérables* : bien entendu, la ressource **GLYC2** implique **GLYC1**, donc il y a 8 paramètres non opérables : $K_{ATP, GLYC2}$, $K_{ATP, GLYC2 PHOX}$, $K_{ATP, GLYC2 LBP}$, $K_{ATP, GLYC2 nLBP}$, $K_{ATP, GLYC2 PHOX LBP}$, $K_{ATP, GLYC2 PHOX nLBP}$, $K_{ATP, GLYC2 LBP nLBP}$ et $K_{ATP, GLYC2 PHOX LBP nLBP}$, laissant au final 24 paramètres à identifier.
2. *Absence d'activateurs* : quand aucun des activateurs (ni **PHOX** ni **GLYC**) n'est présent **ATP** n'active pas ses cibles, qu'il y ait ou non des inhibiteurs et donc $K_{ATP, LBP nLBP} = K_{ATP, LBP} = K_{ATP, nLBP} = K_{ATP} = 0$.
3. *Ensemble de toutes les ressources* : avec l'ensemble de toutes les ressources il n'y a aucune consommation d'ATP et tous les activateurs. Dans ce cas **ATP** est capable d'activer toutes ses cibles et $K_{ATP, GLYC1 GLYC2 LBP nLBP PHOX} = 2$.
4. *Ensemble intermédiaire de ressources* :
 - Toujours dans le cas où il n'y a pas de consommateur de l'ATP et au moins un producteur d'ATP on sait que toutes les cibles de **ATP** vont être activées donc $K_{ATP, LBP nLBP PHOX} = K_{ATP, GLYC1 LBP nLBP PHOX} = K_{ATP, GLYC1 LBP nLBP} = K_{ATP, GLYC1 GLYC2 LBP nLBP} = 2$. A ce point il reste 15 paramètres à identifier.
 - Si **PHOX** (resp. **GLYC** au niveau 2) est présent, il y a suffisamment d'ATP pour activer les voies anaboliques du métabolisme telles que la voie des pentoses phosphates ou les voies de synthèse d'acides aminés, et à inhiber la β -oxydation, mais la production ne sera pas assez puissante pour déclencher les rétro-contrôles de surproduction via **PC** ou **COF**. En effet, la phosphorylation est beaucoup plus productive que la glycolyse en ATP sauf quand le flux glycolytique est très élevé. C'est pourquoi **PHOX** et **GLYC** au niveau 2 permettent l'activation des mêmes cibles en aval de **ATP**. Cela implique que si deux paramètres ne diffèrent que par une ressource et que cette différence n'est due qu'à la présence de **PHOX** dans un paramètre et **GLYC** au niveau 2 dans l'autre, alors ces deux paramètres devront avoir la même valeur. Autrement dit $K_{ATP, \omega GLYC2} = K_{ATP, \omega PHOX}$, où ω est un ensemble de ressources quelconque. De plus, les consommations en ATP provoquées par la production de biomasse (**LBP** et **nLBP** prises ensemble) réduisent la disponibilité en ATP suffisamment pour que l'ATP n'active pas le rétro-contrôle négatif sur **PHOX** via **PC** et donc $K_{ATP, PHOX} = K_{ATP, GLYC1 PHOX} = K_{ATP, GLYC1 GLYC2} = K_{ATP, GLYC1 GLYC2 PHOX} = 1$.
 - Quand **GLYC** au niveau 1 est le seul producteur d'ATP, l'inhibition de **nLBP** et **LBP** empêche **ATP** d'activer ses cibles ($K_{ATP, GLYC1} = 0$).
 - Jusqu'ici 10 paramètres de la variable **ATP** ont encore besoin d'être identifiés : Ceux qui ont comme ressource un producteur d'ATP et exactement un des deux consommateurs d'ATP (**nLBP** ou **LBP**) . Pour aller plus loin dans l'identification nous avons posé l'hypothèse suivante : quand la production de biomasse non lipidique est la seule active, la consommation d'ATP n'est pas suffisante pour réduire la concentration en

ATP et annuler l'activation de **PC**. Avec cette hypothèse on est capable de donner une valeur aux 5 paramètres suivants : $K_{ATP, nLBP PHOX} = K_{ATP, GLYC1 nLBP PHOX} = K_{ATP, GLYC1 GLYC2 nLBP} = K_{ATP, GLYC1 GLYC2 nLBP PHOX} = 1$ and $K_{ATP, GLYC1 nLBP} = 0$.

Comme nous l'avons fait pour 2 paramètres du NADH, nous décidons de laisser 5 paramètres non identifiés pour l'ATP. Les différents graphes de transitions simulant les différentes dynamiques des réseaux instanciés avec les différentes valeurs possibles de ces 7 paramètres seront ensuite validés s'ils satisfont l'ensemble de la matrice de validation proposée en section 9.3. Après avoir formalisé l'ensemble des régulations (graphe de régulation, matrice de validation et identification des paramètres), il reste à créer les fichiers d'input à *GreenBioNet* pour faire la vérification formelle (*via TotemBioNet*) de la cohérence entre, d'une part, le graphe d'interprétation et les paramètres, et d'autre part la matrice de validation.

9.5 Validation de modèles à l'aide de *TotemBioNet*

A ce stade, nous avons établi un graphe de régulation (section 9.2), avec les connaissances statiques des réseaux biochimiques et effectué une identification partielle des paramètres (section 9.4) à l'aide des connaissances sur la dynamique locale. Il s'agit maintenant de les confronter aux phénotypes sur la dynamique globale du système, c.à.d. à la matrice de validation (section 9.3).

La taille de la matrice de validation et le nombre important de contextes différents furent une motivation au développement de *GreenBioNet* au sein de *TotemBioNet* pour obtenir une technique de validation appropriée. Les fichiers se trouvent sur <https://gitlab.com/totembionet/greenbionet> dans les répertoires « metabolism_model ». Nous avons obtenu 15 modèles qui valident l'ensemble des propriétés phénotypiques formalisées dans la matrice ce qui prouve la cohérence du contexte de modélisation décrit dans ce chapitre. Ce contexte intègre donc avec une exhaustivité raisonnable les interactions entre les principales voies métaboliques.

De plus, comme le montre la complexité du chapitre, il n'est pas envisageable de créer de modèles plus grand *ab initio* en se référant à un cycle de développement en "V" décrit dans le chapitre 4. C'est à la suite de l'élaboration de ce modèle que nous avons conclu que l'on devait passer par des étapes de modélisation par enrichissements pour répondre à la problématique de la régulation du métabolisme dans le cancer du pancréas et de son évolution.

Par la suite nous utiliserons donc comme base le contexte de modélisation que nous avons développé dans ce chapitre et nous procéderons à différents enrichissements de ce contexte.

Pour conclure ce chapitre, en suivant la méthode du cycle de vie en « V » ce chapitre propose la formalisation et validation d'un modèle de régulation général du métabolisme de cellule eucaryote exhibant un phénotype particulier : la glycolyse en aérobie. Ce modèle initialement proposé dans [Khoodeeram, 2021] a été le plus grand (en terme de nombre de régulations et de paramètres) conçu avec le formalisme de R.Thomas. D'un point de vue biologique la réalisation de ce modèle est une étape importante car l'ensemble du règne vivant s'appuie sur la régulation du métabolisme cellulaire. Bien entendu ce modèle dit « généraliste », n'est pas utilisable en l'état pour répondre à de nouvelles questions biologiques, car il n'est pas spécifique à chaque type cellulaire. L'intérêt de ce modèle est de pouvoir l'enrichir ou le modifier pour représenter un type cellulaire particulier

par exemple dans le cadre des cellules cancéreuses. La mise en place d'une approche d'enrichissement sur des modèles a déjà été détaillée dans la partie 2 du manuscrit. Les prochains chapitres seront dédiés à l'enrichissement de ce modèle pour la représentation de cellules cancéreuses pancréatiques.

CHAPITRE 10

Enrichissements pour un modèle de cellules cancéreuses pancréatiques non agressives

Ce chapitre résume les différents enrichissements effectués pour la spécification du modèle de la régulation du métabolisme générique à la même problématique dans le cancer du pancréas. L'ensemble des enrichissements effectués sur le modèle se base sur la méthodologie d'enrichissement en spirale décrite dans le chapitre 4 puis développée chapitre 8 du manuscrit.

10.1 Contexte génétique et phénotypique du cancer	141
10.1.1 Concepts biologiques	141
10.1.2 Formalisation du système de régulations pour KRAS	143
10.1.3 Validation par rapport aux connaissances biologiques	147
10.1.4 Prédiction et retour à la biologie	147
10.2 Les cofacteurs $NAD(P)H/NAD(P)^+$ dans le cancer	148
10.2.1 Concepts biologiques	148
10.2.2 Formalisation du système de régulation du NADH	148
10.2.3 Validation par rapport aux connaissances biologiques	152
10.2.4 Prédiction et retour à la biologie	152
10.3 Les ROS comme marqueur de l'état cancéreux	153
10.3.1 Concepts biologiques	153
10.3.2 Formalisation du système de régulation des ROS	154
10.3.3 Validation par rapport aux connaissances biologiques	157
10.3.4 Prédiction et retour à la biologie	158

Pour le développement du modèle de régulation du métabolisme dans le cadre du cancer du pancréas nous nous sommes appuyés sur les connaissances et expériences menées par l'équipe du docteur Sophie Vasseur du Centre de Recherche en Cancérologie de Marseille (CRCM). Cette équipe est spécialisée dans l'étude de l'évolution de l'adénocarcinome canalaire pancréatique (PDAC) vis à vis du métabolisme. Leur but est de comprendre les mécanismes de régulation nécessaires à la mise en place de la maladie, pour ensuite pouvoir concevoir des thérapies nouvelles. Grâce à leurs connaissances, nous avons pu mettre en place une stratégie de modélisation, qui propose trois enrichissements successifs, pour adapter au cancer du pancréas le contexte de modélisation initial de la régulation du métabolisme présenté chapitre 9. Le premier consiste en l'adaptation du modèle au contexte génétique et phénotypique particulier du cancer du pancréas. Le second est un enrichissement lié aux cofacteurs $NADH(P)H$, sur-exprimés dans le cancer et le dernier est lié aux espèces réactives à l'oxygène (ROS) particulièrement représentées dans le PDAC. Pour chacun de ces enrichissements nous suivons la méthodologie incrémentale présentée chapitre 4. Pour chaque enrichissement, nous décrivons les concepts biologiques qui doivent paraître dans le modèle, puis nous décrivons les étapes de formalisation et de validation de ces concepts biologiques dans le modèle (en suivant la méthode décrite chapitre 8). Enfin, pour la dernière étape de chaque enrichissement nous ferons des prédictions pour avoir un retour à la biologie¹.

10.1 Contexte génétique et phénotypique du cancer

Le cancer est une maladie très connue pour ses dérèglements génétiques qui se matérialisent principalement par une division des cellules atteintes de manière désorganisée et constante. L'adénocarcinome canalaire pancréatique (PDAC) est une maladie dévastatrice avec un mauvais pronostic et une incidence croissante. Une détection tardive et une biologie particulièrement agressive sont les éléments majeurs qui déterminent l'échec thérapeutique pour ce type de cancer.

10.1.1 Concepts biologiques

Les mutations génétiques sont responsables de l'apparition et de l'évolution du cancer. Les gènes *KRAS*, *TP53*, *SMAD4*, *CDKN2A*² sont ceux qui sont le plus souvent mutés dans le PDAC [Waters and Der, 2018]. *TP53*, *SMAD4* et *CDKN2A* sont des gènes directement impliqués dans l'augmentation de la division cellulaire puisqu'ils jouent un rôle dans la régulation du cycle cellulaire [Pommier and Kohn, 2003]. Ces mutations font partie des plus fréquentes dans tous les types de cancer. Particulièrement dans le PDAC, plus de 90% des patients portent une mutation délétère sur le gène *KRAS* [Zeitouni et al., 2016]. Un taux aussi élevé de mutation pour ce gène n'est rencontré que dans ce type de cancer : c'est donc une spécificité à intégrer pour placer le modèle dans le contexte du cancer du pancréas.

1. L'ensemble des fichiers permettant la construction de ces enrichissements sont disponibles à l'adresse : <https://gitlab.com/laetitiaG>.

2. *KRAS* : V-Ki-ras2 Kirsten rat sarcoma viral oncogene homolog, *TP53* : tumor protein 53, *SMAD4* : Mothers against decapentaplegic homolog 4 et : cyclin dependent kinase inhibitor 2A.

10.1.1.1 Mutations génétiques de KRAS

KRAS (V-Ki-ras2 Kirsten rat sarcoma viral oncogene homolog) est un gène présent sur le chromosome 12 chez l’humain. Il code pour une GTPases³(enzyme liée au GTP/GDP) qui fait partie de la famille des protéines RAS.

Le fonctionnement de la protéine Kras. Les protéines Kras font partie des voies de signalisation de la cellule qui agissent comme des interrupteurs moléculaires. L’activation et l’inactivation de la protéine KRAS, sont schématisées par la figure 10.1.

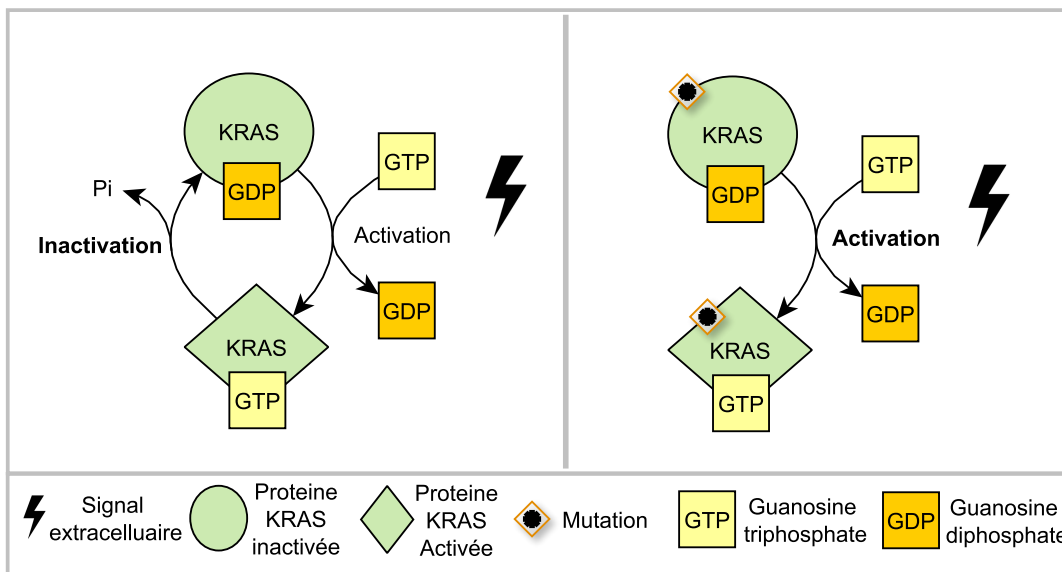


Figure 10.1 – Schéma du fonctionnement de l’interrupteur moléculaire d’une protéine KRAS. Figure adaptée de l’article [Waters and Der, 2018].

Sans signal extracellulaire, la protéine KRAS, dans un cercle vert dans la figure, est liée au GDP (guanosine diphosphate, en jaune orange dans la figure) et est inactive. Lorsque les récepteurs de la cellule sont activés, un signal est lancé. Ce signal provoque l’échange (*via* une enzyme) du GDP associé à la protéine KRAS par un GTP (guanosine triphosphate). Lorsqu’elle est liée à du GTP, la protéine KRAS est activée et transmet à son tour le signal cellulaire par phosphorylation de la cible en aval [Bryant et al., 2014, Storz, 2006]. La phosphorylation de la cible en aval entraîne la transformation du GTP associé à la protéine KRAS en GDP ce qui inactive la protéine (réaction d’inactivation sur la figure).

L’ensemble des étapes d’activation et d’inactivation des protéines KRAS est donné figure 10.1. L’activation n’est pas constante dans le cas d’une cellule saine car le GTP est transformé en GDP, ce qui inactive la protéine KRAS.

3. aussi appelé petite protéine G

Dans le cadre du PDAC, différentes substitutions du gène *KRAS*⁴ peuvent modifier la structure de la protéine *KRAS*, bloquant la transformation du GTP en GDP : la protéine reste donc toujours activée (partie droite de la figure 10.1). Les voies de signalisation en aval de cette protéine sont donc activées constamment ce qui provoque des dérégulations visibles au niveau du phénotype cellulaire.

Conséquences des mutations de *KRAS*. De manière synthétique les protéines *KRAS* médient des signaux de survie, de prolifération, de transformation et d'invasion au niveau de la cellule [Buscaïl et al., 2020]. Ces derniers permettent notamment l'évolution fulgurante du cancer, puisque ces dérégulations se font au niveau du cycle cellulaire. Cependant, ces connaissances ne donnent pas directement d'information sur la régulation du métabolisme dans le cancer, nous devons donc faire appel à d'autres connaissances sur le rôle de *KRAS*.

10.1.1.2 La macropinocytose

La macropinocytose [Andreasson et al., 2021] est un comportement cellulaire observé dans la majorité des cellules cancéreuses dans le pancréas. C'est un processus par lequel des fluides et des nutriments sont ingérés par la cellule, qui replie sa membrane cellulaire vers l'intérieur et forme ainsi une vésicule qui transporte ces nutriments dissous dans la cellule. C'est un processus continu provoqué par l'intermédiaire de la modification des protéines structurales de la membrane (l'actine) elle-même induite par les mutations sur le gène *KRAS*⁵ [Andreasson et al., 2021]. L'abondance et la disponibilité des nutriments sont augmentées et cela a des conséquences mécanistiques sur les différentes voies métaboliques.

10.1.2 Formalisation du système de régulations pour *KRAS*

Nous décidons donc d'ajouter les concepts biologiques liés à la mutation de *KRAS* au modèle de la régulation du métabolisme pour représenter des conditions cancéreuses.

10.1.2.1 Modification du graphe de régulation

Pour l'insertion de la variable *KRAS* nous suivons la méthode d'enrichissement proposée section 8.2.

Nous ajoutons la variable *KRAS* pour représenter le phénomène de macropinocytose. La variable est booléenne. Lorsqu'elle est égale à 0, on considère que le gène *KRAS* n'est pas muté. A l'inverse, lorsque *KRAS*=1 le gène sera muté. La mutation n'est pas directement due à la régulation du métabolisme : la variable *KRAS* n'aura donc pas de régulation entrante. Par contre, la mutation augmente la disponibilité en nutriments qui, dans le modèle à enrichir, est représentée par les variables *AA*, *GLC* et *FA*. En conséquence, la variable *KRAS* doit réguler les trois variables précitées pour les pousser vers leurs niveaux maximum (respectivement 2, 2 et 1, voir section 9.2.1 pour plus d'informations sur les domaines des variables). L'ajout d'une régulation passe dans la

4. Une substitution est une mutation qui remplace un acide aminé présent dans une protéine par un autre. Par exemple, la substitution G12 indique que dans la protéine la glycine (G) en position 12 est substituée par un autre acide aminé qui n'a pas la même fonction. C'est cette mutation qui provoque la dysfonction de protéine *KRAS*.

5. la mutation du gène *KRAS* joue sur la régulation de la production d'actine. L'actine est une protéine de structure qui permet entre autres de jouer sur la morphologie cellulaire. Par l'intermédiaire de la mutation de *KRAS*, l'actine produite modifie la structure de la cellule lui permettant de créer des invaginations au niveau de la membrane cellulaire.

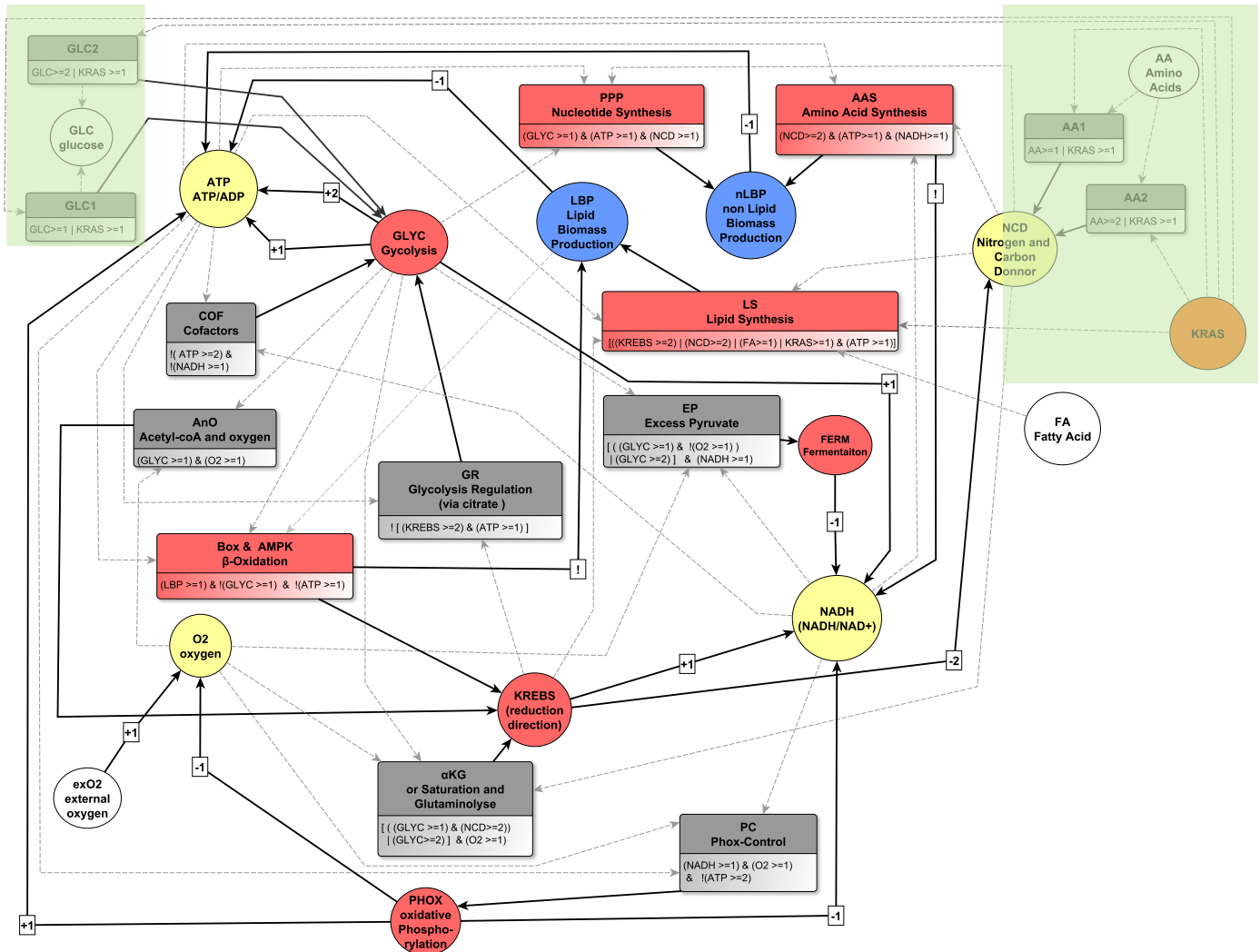


Figure 10.2 – Graphe de régulation enrichi représentant la régulation du métabolisme de la cellule mutée ou non pour le gène KRAS. Les entités bleues font référence à la biomasse, les jaunes aux métabolites et les rouges aux voies métaboliques. L’entité orange fait référence à un composé génétique.

majorité des cas par l’ajout d’un multiplexe. Ici, il existe déjà des multiplexes associés à ces régulations, notés implicitement, qui sont vérifiés lorsque les nutriments sont très fortement disponibles. Ces multiplexes sont représentés par $+1$ et $+2$ entre *GLC* et *GLYC*, $+1$ et $+2$ entre *AA* et *NCD* et explicitement dans *LS* dans la figure 9.1, 110.

- $+1$ entre *GLC* et *GLYC* correspond à un multiplexe dont la formule est $GLC \geq 1$.
- $+2$ entre *GLC* et *GLYC* correspond à un multiplexe dont la formule est $GLC \geq 2$.
- $+1$ entre *AA* et *NCD* indique un multiplexe dont la formule est $AA \geq 1$
- $+2$ entre *AA* et *NCD*, indique un multiplexe dont la formule est $AA \geq 2$.

Nous n'avons donc pas à ajouter de nouveaux multiplexes, mais juste à compléter ceux présentés, car les conditions de régulations proposées par ces multiplexes sont aussi réalisées lorsqu'on a $KRAS=1$.

Les multiplexes $\boxed{+1}$ entre GLC et $GLYC$ et entre AA et NCD doivent eux aussi être modifiés par souci de structure du graphe de régulation. En effet, formellement nous cherchons à faire en sorte que la régulation que $KRAS=1$ exerce, soit similaire à celle cumulée par AA GLC au niveau 2. Or, pour chaque variable d'environnement, si le niveau 2 est atteint et permet d'exercer une régulation, le niveau 1 l'est aussi et exerce sa régulation. C'est pourquoi $KRAS$ au niveau 1 doit être pris en compte dans les régulations $\boxed{+1}$ précitées.

Les formules des multiplexes seront donc modifiées :

La formule ($GLC \geq 1$) du multiplexe $\boxed{+1}$ entre GLC et $GLYC$ devient $GLC \geq 1 | KRAS \geq 1$.
Nous appelons ce multiplexe **GLC1**.

La formule ($GLC \geq 2$) du multiplexe $\boxed{+2}$ entre GLC et $GLYC$ devient $GLC \geq 2 | KRAS \geq 1$.
Nous appelons ce multiplexe **GLC2**.

La formule ($AA \geq 1$) du multiplexe $\boxed{+1}$ entre AA et NCD devient $AA \geq 2 | KRAS \geq 1$.
Nous appelons ce multiplexe **AA1**.

La formule ($AA \geq 2$) du multiplexe $\boxed{+2}$ entre AA et NCD devient $AA \geq 2 | KRAS \geq 1$.
Nous appelons ce multiplexe **AA2**.

Enfin, FA est déjà impliqué dans un multiplexe, pour prendre en compte la régulation par $KRAS$, il suffit d'ajouter à la formule la possibilité de remplacer $FA \geq 1$ par $KRAS \geq 1$. La formule du multiplexe **LS** est donc remplacée par

$$(KREBS \geq 2 | NCD \geq 2 | FA \geq 1 | KRAS \geq 1) \& ATP \geq 1$$

Au final on obtient le graphe de régulation donné figure 10.2.

10.1.2.2 Modification cohérente des paramètres

Comme nous n'avons pas ajouté de multiplexe, nous n'avons ajouté aucune nouvelle ressource aux variables. Il n'y a donc aucun nouveau paramètre. De plus comme nous savons que les modifications dues à la mutation du gène $KRAS$ ont le même impact que les régulations déjà présentes, le sens abstrait de multiplexes n'est pas modifié ce qui ne remet pas en cause l'identification des paramètres précédant à l'enrichissement : nous n'avons donc aucun paramètre à réévaluer.

10.1.2.3 Modification de la matrice de validation

La variable $KRAS$ n'est pas régulée par d'autres éléments du graphe : c'est une variable d'environnement. On doit donc ajouter une colonne dans la matrice de validation pour cette variable. Comme la variable peut prendre deux valeurs, le nombre de lignes sera à priori doublé. Sans mutation du gène $KRAS$, lorsque $KRAS=0$, nous sommes dans les conditions générales de la régulation du métabolisme. L'ensemble de la matrice donnée table 9.1 peut donc être reporté. Cela constitue la partie de la matrice qui contribue à la non régression des propriétés préalablement établies. Le plus complexe est de définir les comportements lorsque $KRAS=1$. En premier lieu, on peut factoriser de nombreuses lignes car la structure du graphe de régulation implique que lorsque $KRAS=1$, peu importe l'état réel de GLC , AA et FA , les multiplexes permettant d'activer les cibles des variables d'environnement. En effet par défaut, $KRAS=1$ fixe les apports en acides aminés, sucres, et lipides à leur maximum. En conséquence, peu importe la valeur initiale de GLC , AA

Environnements	KRAS	FA	exO2	GLC	AA	ATP (0-2)	O2 (0-1)	GLYC (0-2)	nLBP (0-1)	LBP (0-1)	FERM (0-1)	KREBS (0-2)	PHOX (0-1)	NADH (0-1)	NCD (0-2)
Sans Mutations															
absence de lipides et d'oxygène															
1	0	0	0	0	0	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)
2	0	0	0	0	1	osc	td(0)								
3	0	0	0	0	2		td(0)								
4	0	0	0	1	0	!td(0)	td(0)	osc(0-1)			!td(0)		td(0)	osc	td(0)
5	0	0	0	1	1	osc	td(0)	osc(0-1)	!td(0)		!td(0)		td(0)	osc	td(1)
6	0	0	0	1	2	osc	td(0)	osc	!td(0)	!td(0)	!td(0)		td(0)	osc	td(2)
7	0	0	0	2	0	!td(0)	td(0)	osc			!td(0)		td(0)	osc	td(0)
8	0	0	0	2	1	osc	td(0)	osc	!td(0)		!td(0)		td(0)	osc	td(1)
9	0	0	0	2	2	osc	td(0)	osc	!td(0)	!td(0)	!td(0)		td(0)	osc	td(2)
Pas de lipides mais apport d'oxygène															
10	0	0	1	0	0	td(0)	!td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)
11	0	0	1	0	1		!td(0)								
12	0	0	1	0	2		!td(0)								
13	0	0	1	1	0	!td(0)	!td(0)	osc			td(0)	osc	osc	osc	td(0)
14	0	0	1	1	1	osc	!td(0)	osc	!td(0)		td(0)	osc	osc	osc	td(1)
15	0	0	1	1	2	osc	!td(0)	osc	!td(0)	!td(0)	td(0)	osc	osc	osc	td(2)
16	0	0	1	2	0	!td(0)	!td(0)	osc			!td(0)		!td(1)	osc	td(0)
17	0	0	1	2	1	osc	!td(0)	osc	!td(0)	!td(0)	!td(0)		!td(1)	osc	td(1)
18	0	0	1	2	2	osc	!td(0)	osc	!td(0)	!td(0)	!td(0)		!td(1)	osc	td(2)
Présence de lipides mais pas d'apport en oxygène															
19	0	1	0	0	0	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)
20	0	1	0	0	1		td(0)								
21	0	1	0	0	2		td(0)								
22	0	1	0	1	0	osc	td(0)	osc(0-1)		!td(0)	!td(0)		td(0)	osc	td(0)
23	0	1	0	1	1	osc	td(0)	osc(0-1)	!td(0)	!td(0)	!td(0)		td(0)	osc	td(1)
24	0	1	0	1	2	osc	td(0)	osc	!td(0)	!td(0)	!td(0)		td(0)	osc	td(2)
25	0	1	0	2	0	osc	td(0)	osc		!td(0)	!td(0)		td(0)	osc	td(0)
26	0	1	0	2	1	osc	td(0)	osc	!td(0)	!td(0)	!td(0)		td(0)	osc	td(1)
27	0	1	0	2	2	osc	td(0)	osc	!td(0)	!td(0)	!td(0)		td(0)	osc	td(2)
Présence de lipides et d'oxygène															
28	0	1	1	0	0	td(0)	!td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)	td(0)
29	0	1	1	0	1		!td(0)								
30	0	1	1	0	2		!td(0)								
31	0	1	1	1	0	osc	!td(0)	osc		!td(0)	td(0)	osc	osc	osc	td(0)
32	0	1	1	1	1	osc	!td(0)	osc	!td(0)	!td(0)	td(0)	osc	osc	osc	td(1)
33	0	1	1	1	2	osc	!td(0)	osc	!td(0)	!td(0)		osc	osc	osc	td(2)
34	0	1	1	2	0	osc	!td(0)	osc		!td(0)	!td(0)		!td(1)	osc	td(0)
35	0	1	1	2	1	osc	!td(0)	osc	!td(0)	!td(0)	!td(0)		!td(1)	osc	td(1)
36	0	1	1	2	2	osc	!td(0)	osc	!td(0)	!td(0)	!td(0)		!td(1)	osc	td(2)
Mutation KRAS															
37	1		0												
38	1		1												

Table 10.1 – Matrice de validation pour le modèle de la régulation du métabolisme cellulaire avec ou sans mutation du gène KRAS. Les lignes ~1 à ~36 correspondent aux lignes déjà exhibées dans le modèle général de la régulation du métabolisme. Les lignes ~37 et ~38 sont de nouvelles spécifications. Nous avons des cases grises dans l'environnement pour stipuler que la valeur des variables d'environnement pour ces cases n'est pas importante.

ou *FA*, le modèle agira de la même manière. Seul le taux d'oxygène peut continuer à varier, et jouer sur le comportement du modèle : il y aura donc seulement autant de nouvelles lignes où *KRAS*=1 que de niveaux à l'oxygène. Comme la variable *ExO2* est booléenne, on ajoute donc

à la matrice finale deux lignes (où les cases décrivant l'état des variables d'environnement **GLC**, **AA** ou **FA** sont grisées car non déterminantes). Dans ces deux lignes, nous n'indiquerons pas de nouveaux comportements, même si nous avons des connaissances potentielles au préalable sur les comportements, pour voir si la mutation indiquée nous amène à de nouveaux comportements inconnus.

10.1.3 Validation par rapport aux connaissances biologiques

Maintenant que nous avons construit un contexte de modélisation enrichi, nous suivons la méthodologie développée au chapitre 4 en validant le contexte de modélisation enrichi *via* le module *GreenBioNet* de *TotemBioNet*. Comme nous n'avons pas ajouté de paramètre, le nombre de paramétrisations à énumérer pour *TotemBioNet* est le même que lors de l'élaboration du modèle de la régulation du métabolisme général (15 paramétrisations, voir chapitre 9). toutes les paramétrisations énumérées sont validées. Comme nous n'avons pas ajouté de comportement par rapport au modèle initial dans cet enrichissement, la validation pour cet enrichissement ne sert que de test de non régression.

10.1.4 Prédiction et retour à la biologie

Une fois un ensemble de modèles validé, il reste la dernière étape : la prédiction et le retour à la biologie.

Lorsqu'on regarde en détail les comportements prédits, donnés table 10.2, on constate que pour tous les modèles validés, les comportements des lignes $\rightsquigarrow 37$ et $\rightsquigarrow 38$ nouvellement ajoutées sont similaires à ceux des $\rightsquigarrow 27$ et $\rightsquigarrow 36$ de la table 10.1, qui représentent des phénotypes de glycolyse en aérobie.

<i>Environnements</i>	KRAS	FA	exO2	GLC	AA	ATP (0-2)	O2 (0-1)	GLYC (0-2)	nLBP (0-1)	LBP (0-1)	FERM (0-1)	KREBS (0-2)	PHOX (0-1)	NADH (0-1)	NCD (0-2)
Mutation KRAS															
37	1		0			osc	!td(0)	osc	!td(0)	!td(0)	!td(0)	osc	td(0)	osc	td(2)
38	1		1			osc	!td(0)	osc	!td(0)	!td(0)	!td(0)	osc	!td(1)	osc	td(2)

Table 10.2 – Matrice de prédiction pour les lignes $\rightsquigarrow 37$ et $\rightsquigarrow 38$.

Les connaissances biologiques nous indiquent qu'une cellule cancéreuse dans le pancréas a un phénotype de glycolyse en aérobie [Cao et al., 2020]. Il est donc normal d'obtenir ces résultats. Les résultats de ces lignes indiquent que le métabolisme de la cellule est augmenté vis à vis d'une cellule dans des conditions normales (données par la ligne $\rightsquigarrow 14$ de la table 10.1).

On peut donc dire que la mutation KRAS a été modélisée correctement et que les régulations ajoutées dans l'enrichissement sont suffisantes pour démontrer que la cellule cancéreuse dans le pancréas utilise la respiration aérobie.

Comme tous les modèles exhibent les mêmes comportements asymptotiques et que leur nombre est suffisamment restreint, nous décidons de les conserver. Nous n'effectuons pas de restrictions sur les paramètres non identifiés. Pour le prochain enrichissement incrémental nous repartirons donc du contexte de modélisation créé dans cette section sans restriction de paramétrisation.

Ce premier enrichissement nous a permis de nous placer dans un contexte cancéreux. Cependant, d'un point de vue phénotypique, nous n'observons pas de différences entre les lignes

porteuses de la mutation $\rightsquigarrow 37$ et $\rightsquigarrow 38$ et les lignes $\rightsquigarrow 27$ et $\rightsquigarrow 36$ (seulement pathologiques). Nous devons donc faire un travail sur les marqueurs et modifications connues biologiquement pour pouvoir caractériser dans le modèle ces différentes conditions.

10.2 Les cofacteurs $NAD(P)H/NAD(P)^+$ dans le cancer

Dans le cancer, les cofacteurs NADH et le NADPH jouent des rôles essentiels. Le NADH et le NADPH sont impliqués dans de nombreuses voies métaboliques qui sont dérégulées dans le cancer. Il convient donc d'étudier les concepts biologiques associés à ces cofacteurs pour ensuite pouvoir formaliser ces connaissances dans le modèle.

10.2.1 Concepts biologiques

10.2.1.1 Le ratio $NADH/NAD^+$ et le cancer

Comme nous l'avons déjà présenté dans le chapitre 9, page 115, le NAD^+ , est un cofacteur qui médie les réactions d'oxydoréduction dans diverses voies métaboliques, comme la glycolyse, le cycle de KREBS ou la phosphorylation oxydative. Une fois utilisé, il bascule sous la forme $NADH$. Dans le contexte du cancer, le phénotype d'aérobie glycolytique de la cellule permet le réapprovisionnement du cofacteur. Des niveaux plus élevés que dans les cellules saines ont été retrouvés. Indirectement, la mutation KRAS est impliquée dans l'augmentation du ratio $NADH/NAD^+$, via la régulation des voies métaboliques [Yaku et al., 2018].

10.2.1.2 Le ratio $NADPH/NADP^+$ et le cancer

De manière similaire le $NADPH$ ne diffère chimiquement du $NADH$ que par un groupement phosphate. Il médie aussi des réactions d'oxydoréductions plutôt pour des voies anaboliques. Cependant, ce cofacteur est plus connu pour son implication dans la régulation et le maintien de la balance red/ox⁶ de la cellule. Nous reparlerons de cette balance dans l'enrichissement suivant. Le ratio $NADPH/NADP^+$ est aussi augmenté dans le PDAC. Ce ratio est augmenté car certaines enzymes telles que GOT1⁷ et ME1⁸ sont sur-exprimées lorsque KRAS est muté. Ces enzymes appartiennent au métabolisme des acides aminés et plus précisément celui de la glutamine [Son et al., 2013].

10.2.2 Formalisation du système de régulation du NADH

Dans le modèle les ratio de $NADH/NAD^+$ et $NADPH/NADP^+$ sont formalisés par la variable $NADH$. Sachant que les deux ratios sont augmentés dans le PDAC et que ces modifications sont causées directement par les mutations de KRAS nous pouvons continuer de travailler sur cette variable pour définir plus précisément les cellules cancéreuses dans le pancréas. Dans cette section nous procéderons à deux enrichissements simultanés. Le premier consiste à ajouter un multiplexe pour représenter la régulation particulière du métabolisme de la glutamine et le

6. équilibre des espèces chimiques oxydatives et réductrices dans la cellule ; cet équilibre est très important pour la régulation du métabolisme cellulaire.

7. glutamic-oxaloacetic transaminase 1, amino-transférase humaine

8. NADP-dependent malic enzyme 1

second à ajouter un niveau à la variable $NADH$, pour pouvoir spécifier différents phénotypes en fonction de l'état de la cellule (cancéreux ou non).

10.2.2.1 Modifications du graphe de régulation

Ajout du multiplexe GLN *metabolism*. Nous considérons que les modifications indirectes du ratio $NADH/NAD^+$ sont déjà représentées par les chemins existants du graphe de régulation. Nous nous intéressons donc à la régulation du ratio $NADPH/NADP^+$ dans le cancer du pancréas *via* la régulation du métabolisme des acides aminés. Nous ajoutons un multiplexe appelé **GLN *metabolism***⁹ qui régule la variable $NADH$. La régulation est effective, lorsqu'il y a suffisamment de glutamine dans la cellule et que le gène $KRAS$ est muté ($KRAS \geq 1$). La glutamine présente dans la cellule est représentée par la variable NCD . La formule que l'on en déduit pour le multiplexe **GLN *metabolism*** est par conséquent : $KRAS \geq 1 \& NCD \geq 1$. Le multiplexe est représenté dans la version enrichie du graphe de régulation donné figure 10.3.

Réévaluation des niveaux de la variable $NADH$. La variable $NADH$ dans le modèle initial présenté chapitre 9 est booléenne et n'a pas été modifiée lors de l'enrichissement décrit dans la section précédente. Dans ce paragraphe nous formalisons un niveau supplémentaire pour $NADH$. Pour gagner en expressivité il faut pouvoir faire la différence entre les cellules non cancéreuses et les cellules cancéreuses. Nous choisissons d'ajouter un niveau entre le niveau 0 et le niveau 1. Le sens du niveau 0 n'est pas modifié par rapport au chapitre 9 : le niveau de $NADH$ est trop faible pour activer les cibles en aval. Le niveau 1 intermédiaire correspond à un ratio $NADPH/NADP^+$ et $NADH/NAD^+$ moyen. Nous considérons à ce stade de l'enrichissement que le niveau 1 n'est pas suffisant pour activer les cibles. Le niveau 2 correspond à un niveau élevé pour $NADH$, qui lui est capable d'activer ses cibles (anciennement niveau 1).

Remarque – En ne considérant que cet enrichissement, l'ajout d'un niveau pour représenter le $NADH$ peut paraître peu utile si l'on suit le principe du rasoir d'Ockham[Sober, 2015]¹⁰, car le profil d'activation des cibles du $NADH$ n'est pas modifié entre le niveau 0 et 1. Cependant, cet enrichissement s'inscrit dans un cycle d'enrichissements volontairement limité à des tailles raisonnables. Le $NADH$ étant un pilier central dans le métabolisme cellulaire le passage par cette modification sans pour autant appliquer d'autres enrichissements simultanés est important pour faciliter les preuves de non-régression des comportements déjà obtenus.

La deuxième étape de l'ajout d'un niveau consiste à revoir tous les seuils sortants de la variable pour laquelle nous venons d'ajouter un niveau. Comme nous avons indiqué que le niveau 2 est le seul capable d'activer les cibles de $NADH$, tous les multiplexes cibles de la variable voient donc leur formule modifiée : la partie de la formule notée $NADH \geq 1$ des multiplexes **PC**, **AAS**, **EP** devient $NADH \geq 2$ et la partie $!NADH \geq 1$ du multiplexe **cof** devient $!NADH \geq 2$. L'ensemble des modifications apportées au graphe pour construire cet enrichissement est donné en figure 10.3.

9. GLN correspond à l'abréviation 3 lettre de l'acide aminée glutamine.

10. Aussi connu sous le nom de principe de parcimonie qui propose de n'utiliser que le minimum de causes élémentaires pour expliquer un phénomène

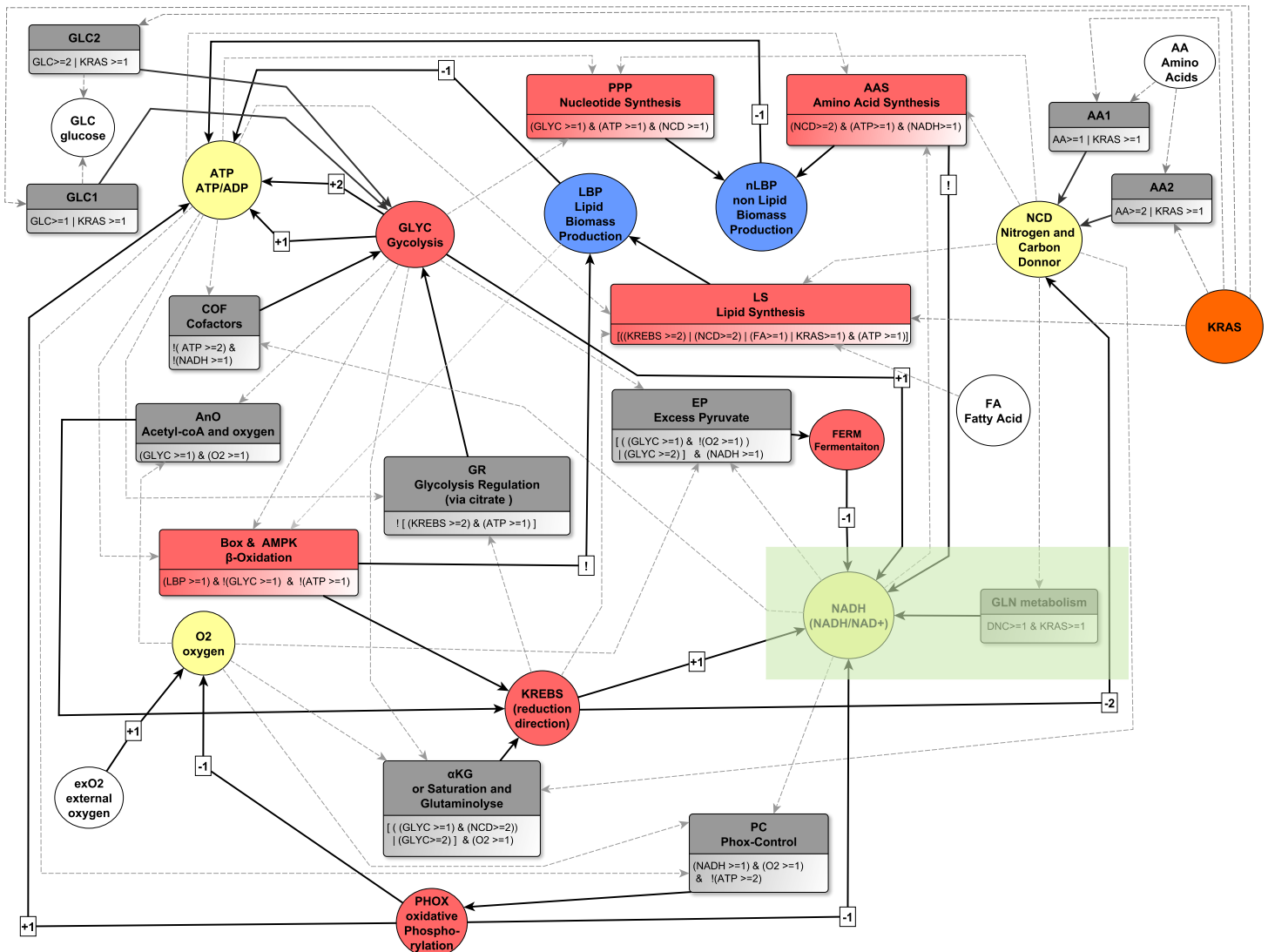


Figure 10.3 – Graphe de régulation enrichi avec le multiplexe **GLN metabolism** et le changement des seuils pour les multiplexes impliquant la variable **NADH** (encadré vert). Les entités bleues font référence à la biomasse, les jaunes aux métabolites et les rouges aux voies métaboliques. L'entité orange fait référence à un composé génétique.

10.2.2.2 Modification cohérente des paramètres de NADH.

Nous avons 32 paramètres avant enrichissement, la formalisation du multiplexe **GLN metabolism** ajoute une ressource à la variable **NADH** : Cela double le nombre de paramètres (soit 64). 32 des 64 paramètres de **NADH** sont déjà présents et ont été identifiés dans le modèle initial (voir 9.2, page 128). Tous les paramètres qui existaient dans le modèle initial et qui étaient valués à 0 peuvent le rester car le sens du niveau 0 n'est pas modifié. Pour tous ceux qui dans le modèle initial sont estimés à 1, le sens est modifié. Lorsque les paramètres valaient 1 dans le

modèle initial, toutes les cibles étaient activées. La raison biologique en a été explicitée dans le chapitre 9, et si l'on souhaite conserver le sens qui a été donné, le nouveau niveau 2 correspond en tous points au niveau 1 de l'ancien modèle. On peut donc porter à 2 la totalité des paramètres initialement à 1. Deux paramètres restent non identifiés dans le modèle initial ; par transitivité ces paramètres ne peuvent toujours pas être identifiés car aucune nouvelle information biologique ne donne d'information. Grâce à la conservation et l'adaptation du précédent modèle nous avons estimés la moitié des paramètres associé au $NADH$ soit 32 paramètres (cf. table 10.3) : l'estimation des paramètres n'est donc pas terminée,

$K_{NADH, GLYC KREBS AAS PHOX} = 2$	$K_{NADH, KREBS AAS} = 0$	$K_{NADH} = 0$
$K_{NADH, GLYC KREBS AAS PHOX} = 2$	$K_{NADH, KREBS} = 0$	$K_{NADH, FERM} = 0$
$K_{NADH, GLYC KREBS AAS PHOX} = 2$	$K_{NADH, GLYC KREBS} = 0$	$K_{NADH, PHOX} = 0$
$K_{NADH, FERM GLYC KREBS PHOX} = 2$	$K_{NADH, FERM GLYC} = 0$	$K_{NADH, AAS} = 0$
$K_{NADH, GLYC KREBS FERMAAS PHOX} = 2$	$K_{NADH, GLYC} = 0$	$K_{NADH, GLYC KREBS AAS} = 0$
$K_{NADH, KREBS FERMPHOX} = 2$	$K_{NADH, FERM GLYC AAS} = 0$	$K_{NADH, FERMAAS} = 0$
$K_{NADH, KREBS PHOX} = 2$	$K_{NADH, GLYC PHOX} = 0$	$K_{NADH, AAS PHOX} = 0$
$K_{NADH, GLYC KREBS PHOX} = 2$	$K_{NADH, FERMPHOX} = 0$	$K_{NADH, FERMAAS PHOX} = 0$
$K_{NADH, KREBS FERMAAS PHOX} = 2$	$K_{NADH, GLYC AAS} = 0$	$K_{NADH, FERM KREBS AAS} = 0$
$K_{NADH, GLYC FERMAAS PHOX} = 2$	$K_{NADH, FERM KREBS} = 0$	$K_{NADH, GLYC AAS PHOX} = 0$
$K_{NADH, FERM GLYC PHOX}$	$K_{NADH, FERM GLYC KREBS}$	

Table 10.3 – Paramètres ré-estimés à partir du sens des paramètres du modèle initial après dédoublement du niveau de $NADH$.

A ce point nous devons encore identifier les 32 paramètres pour lesquels la ressource **GLN metabolism** est présente. D'après les conditions de Snoussi tous les paramètres qui figurent dans le tableau 10.3 et qui ont pour valeur 2 auxquels on ajoute une ressource seront égaux à 2 (ces paramètres correspondent à la colonne de gauche du tableau 10.4). Concernant les autres paramètres nous savons que lorsque la ressource « $GLNm$ » est présente nous sommes dans un cadre cancéreux et le niveau des cofacteurs est supérieur au niveau que l'on a lorsque la cellule n'est pas cancéreuse. En comparaison, tous les paramètres qui sont à 0 dans le tableau 10.3 sans la ressource sont maintenant, avec la ressource, égaux à 1 (voir table 10.4).

$K_{NADH, GLYC KREBS AAS PHOX GLNm} = 2$	$K_{NADH, KREBS AAS GLNm} = 1$	$K_{NADH, GLNm} = 1$
$K_{NADH, GLYC KREBS AAS PHOX GLNm} = 2$	$K_{NADH, KREBS GLNm} = 1$	$K_{NADH, FERM GLNm} = 1$
$K_{NADH, GLYC KREBS AAS PHOX GLNm} = 2$	$K_{NADH, GLYC KREBS GLNm} = 1$	$K_{NADH, PHOX GLNm} = 1$
$K_{NADH, FERM GLYC KREBS PHOX GLNm} = 2$	$K_{NADH, FERM GLYC GLNm} = 1$	$K_{NADH, AAS GLNm} = 1$
$K_{NADH, GLYC KREBS FERMAAS PHOX GLNm} = 2$	$K_{NADH, GLYC GLNm} = 1$	$K_{NADH, GLYC KREBS AAS GLNm} = 1$
$K_{NADH, KREBS FERMPHOX GLNm} = 2$	$K_{NADH, FERM GLYC AAS GLNm} = 1$	$K_{NADH, FERMAAS GLNm} = 1$
$K_{NADH, KREBS PHOX GLNm} = 2$	$K_{NADH, GLYC PHOX GLNm} = 1$	$K_{NADH, AAS PHOX GLNm} = 1$
$K_{NADH, GLYC KREBS PHOX GLNm} = 2$	$K_{NADH, FERMPHOX GLNm} = 1$	$K_{NADH, FERMAAS PHOX GLNm} = 1$
$K_{NADH, KREBS FERMAAS PHOX GLNm} = 2$	$K_{NADH, GLYC AAS GLNm} = 1$	$K_{NADH, FERM KREBS AAS GLNm} = 1$
$K_{NADH, GLYC FERMAAS PHOX GLNm} = 2$	$K_{NADH, FERM KREBS GLNm} = 1$	$K_{NADH, GLYC AAS PHOX GLNm} = 1$

Table 10.4 – Nouveaux paramètres de la variable $NADH$ dûs à l'ajout du multiplexe **GLN metabolism**.

Enfin, biologiquement nous n'avons aucune indication sur la dynamique locale pour les paramètres $K_{NADH, FERM GLYC PHOX GLNm}$ et $K_{NADH, FERM GLYC KREBS GLNm}$. Au final il reste 4 paramètres non identifiés pour la variable $NADH$ ($K_{NADH, FERM GLYC PHOX}$, $K_{NADH, FERM GLYC KREBS}$, $K_{NADH, FERM GLYC PHOX GLNm}$ et $K_{NADH, FERM GLYC KREBS GLNm}$).

10.2.2.3 Modification de la matrice de validation

La matrice de validation est peu modifiée, en effet, nous n'avons eu d'impact que sur la variable *NADH*, nous savons que globalement dans les lignes où *KRAS* = 1 (\rightsquigarrow 37 et \rightsquigarrow 38) le comportement de *NADH* doit être différent de celles où l'on a tous les nutriments disponibles (acides gras, sucres, acides aminés) lignes \rightsquigarrow 16 et \rightsquigarrow 36. Comme nous ajoutons un niveau à la variable, tous les comportements de la colonne sont à réévaluer. Par défaut, l'ajout d'un niveau ne change pas le niveau 0, ceci implique que pour toutes les lignes où *NADH* tend vers 0, (\rightsquigarrow 1, \rightsquigarrow 10, \rightsquigarrow 19, \rightsquigarrow 28) le comportement peut être conservé malgré l'enrichissement. Pour toutes les cases indiquant le comportement « osc », par défaut si l'on souhaite la plus faible régression possible nous devons conserver le pattern. Cependant l'ajout d'un niveau élargit les comportements possibles. En effet, avant nous savions que *NADH* oscillait entre 0 et 1 car la variable était booléenne. Maintenant, dans la matrice enrichie vis à vis de la variable *NADH*, l'oscillation pourra être soit entre 0 et 1, 1 et 2 ou 0 et 2. Enfin, pour les lignes cancéreuses, \rightsquigarrow 37 et \rightsquigarrow 38 aucune indication ne nous permet de compléter plus la matrice de validation. Au final, nous obtenons la même matrice de validation que pour l'enrichissement précédent (voir 10.1 pour les lignes \rightsquigarrow 1 à \rightsquigarrow 36) à ceci près que dans la première ligne indiquant le domaine de la variable *NADH* nous avons (0-2) à la place de (0-1). Concernant les lignes cancéreuses (\rightsquigarrow 1 à \rightsquigarrow 36) nous reprenons l'ensemble des comportements prédits pour les variables dans le cadre de l'enrichissement du modèle avec *KRAS*, mais nous enlevons les comportements liés au *NADH*, de cette manière on conserve au maximum les comportements prédits pendant l'enrichissement précédent. La version finale de la matrice est donnée table 10.5.

Environnements	KRAS	FA	exO2	GLC	AA	ATP (0-2)	O2 (0-1)	GLYC (0-2)	nLBP (0-1)	LBP (0-1)	FERM (0-1)	KREBS (0-2)	PHOX (0-1)	NADH (0-2)	NCD (0-2)
Sans Mutations															
1 à 36	0	0 & 1	0 & 1	0 & 1 & 2	0 & 1 & 2	Identique matrice modèle donnée 10.1									
Mutation KRAS															
37	1		0			osc	td(0)	osc	!td(0)	!td(0)	!td(0)	osc	td(0)		td(2)
38	1		1			osc	!td(0)	osc	!td(0)	!td(0)	!td(0)	osc	!td(1)		td(2)

Table 10.5 – Matrice de validation pour le modèle de la régulation du métabolisme avec ajout du multiplexe **GLN metabolism** et l'ajout d'un niveau à la variable *NADH*. Les lignes \rightsquigarrow 1 à \rightsquigarrow 36 correspondent aux lignes déjà exhibées dans le modèle général de la régulation du métabolisme. Les lignes \rightsquigarrow 37 et \rightsquigarrow 38 sont de nouvelles spécifications.

10.2.3 Validation par rapport aux connaissances biologiques

Pour l'enrichissement autour du *NADH* *TotemBioNet* a permis de valider 6 paramétrisations différentes. En analysant les résultats pour valider les comportements demandés, deux éléments sont à mettre en avant. Premièrement les paramètres K_{NADH} , $FERM_{GLYC}$ $PHOX$ $GLNm$ et K_{NADH} , $FERM_{GLYC}$ $PHOX$ doivent avoir la valeur 2. Ceci permet de donner des informations aux biologistes sur l'évolution locale des régulations.

10.2.4 Prédiction et retour à la biologie

En regardant en détail les comportements prédits on peut vérifier que les modèles cancéreux ont bel et bien des comportements différents mais que le niveau des cofacteurs n'est pas constant, qu'il oscille. Lorsque *KRAS*=0, dans les environnements où *NADH* oscille, l'oscillation se fait

entre 0 et 2. A l'inverse dans les lignes où $KRAS=1$, l'oscillation se fait entre 1 et 2. Cette nuance permet d'expliquer le niveau observé biologiquement, malgré l'oscillation le niveau basal moyen pour $NADH$ est plus élevé dans le cancer. Grâce à ces prédictions, nous pouvons affirmer qu'une mutation du gène $KRAS$ est suffisante pour obtenir une augmentation des ratio de cofacteurs dans la cellule.

Hormis la contextualisation au cancer *via* la variable $KRAS$, nous avons enrichi le modèle en travaillant sur la modifications d'éléments déjà présents. Cependant les connaissances sur le métabolisme cancéreux dans le pancréas ne s'arrête pas là. En effet, l'état red/ox de la cellule visualisable par la quantité de ROS présente dans la cellule est un marqueur très étudié dans le cancer du pancréas. Nous allons donc procéder à un dernier enrichissement pour représenter explicitement ces espèces moléculaire dans le modèle.

10.3 Les ROS comme marqueur de l'état cancéreux

10.3.1 Concepts biologiques

Les ROS regroupent plusieurs types de molécules qui ont la particularité commune d'être réactives à l'oxygène d'où leur nom : Reactive Oxygen Species.

10.3.1.1 Les espèces réactives à l'oxygène

L'accumulation de ces espèces crée des dommages à l'ADN hautement délétères pour la cellule. Lorsque la cellule non cancéreuse ne gère plus la quantité de ROS, elle passe en apoptose (mort cellulaire programmée). La régulation de leur quantité est donc cruciale. La cellule a différents mécanismes pour réduire ces espèces délétères. Paradoxalement, c'est la cellule elle même qui crée ces espèces *via* des processus métaboliques nécessaires tels que la phosphorylation oxydative. En effet la mécanique moléculaire n'est pas totalement fiable car près de 1% du fonctionnement de cette voie produit des ROS. Il a été démontré que le taux de ces espèces est bien plus haut dans les cellules cancéreuses que dans les cellules saines particulièrement dans le cancer du pancréas [Martinez-Useros et al., 2017].

10.3.1.2 Description des mécanismes de création et de réduction des ROS

La source des ROS dans la cellule. Les ROS peuvent être créés par la phosphorylation oxydative. Par de nombreuses réactions d'oxydoréduction, cette voie permet la production d'ATP. C'est cette chaîne qui provoque la production de ROS. Un autre producteur de ROS intracellulaire est la NADPHoxydase, une enzyme plus connue sous le nom de NOX. L'enzyme catalyse une réaction qui permet, à partir de $NADPH$ et d'oxygène, de produire du NAD^+ et des ROS [Vaquero et al., 2004, Purohit et al., 2019]. D'autres mécanismes *via* le peroxydase et la peroxydation des lipides existent, mais ces mécanismes sont beaucoup moins fréquents, nous ne les prendrons donc pas en compte.

Il existe plusieurs NOX homologues et certains tels que NOX4 sont plus exprimés dans le cancer du pancréas, ce qui explique en partie l'augmentation de la quantité de ROS dans la cellule cancéreuse [Bi et al., 2021]. Dans l'article [Martinez-Useros et al., 2017] les auteurs indiquent que dans le PDAC, les NOX représentent la source majeure de production de ROS.

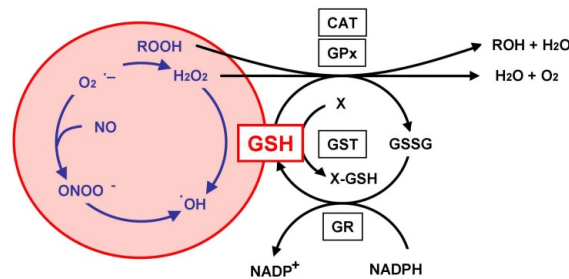


Figure 10.4 – Réactions de détoxification des ROS par l'intermédiaire du Couple GSH/GSSH. Figure provenant de l'article [Aoyama and Nakaki, 2012].

La détoxification des ROS dans la cellule. La réduction du taux de ROS de la cellule se fait par l'intermédiaire de la GST (glutathion S-transférase) qui fait appel au glutathion (voir figure 10.4). Le glutathion est un peptide de trois acides aminés, qui existe sous formes oxydée (GSH) et réduite (GSSH). Par l'intermédiaire d'une réaction chimique la GSH est oxydée et le ROS est détoxifié pour former de l'eau. Pendant cette réaction le GSH devient du GSSH. Le pool de GSH est ensuite régénéré par l'intermédiaire de l'enzyme GR (Glutathion Réductase). Cette réaction nécessite du NADPH. D'autres enzymes comme la SOD (Superoxyde Dismutase) ou la GPx (Peroxyrédoxine) permettent aussi la détoxification cellulaire, mais nous ne rentrerons pas dans les détails de leur fonctionnement car ils sont minoritaires.

10.3.2 Formalisation du système de régulation des ROS

Les ROS étant un marqueur très étudié dans le cadre de PDAC et de son évolution, ces espèces ont une place importante dans le modèle que nous mettons en place. Comme ROS ne sont pas présents dans le modèle, même implicitement, nous choisissons de l'ajouter en temps que nouvelle variable dans le but de distinguer les cellules saines des cellules cancéreuses par un marqueur spécifique. La variable **ROS** représente donc tous les composés réactifs à l'oxygène jouant un rôle important dans la balance red/ox de la cellule. Cette variable aura 3 niveaux ; le niveau 0 qui indique un niveau sain de ROS dans la cellule, le niveau 1 qui correspond à un niveau élevé de ROS gérable par la cellule cancéreuse notamment, et le niveau 2 qui correspond à un niveau tel que la cellule par mécanisme d'autodéfense entre en apoptose.

10.3.2.1 Modification du graphe de régulation

Nous allons lier au modèle la nouvelle variable (**ROS**) en commençant par ses régulations en aval, comme l'indique la méthodologie d'enrichissement de modèles décrite section 4.2.1. Les conséquences biologiques associées aux ROS ont malheureusement un spectre diffus. Biologiquement, nous ne savons pas exactement si les ROS ont un impact sur le métabolisme ou non. Par défaut pour cet enrichissement nous utiliserons cette variable comme marqueur de l'état cellulaire, mais nous ne l'utiliserons pas pour produire de nouvelles hypothèses de régulations. Cette variable n'aura donc pas *à priori* de régulations en aval.

A l'inverse pour établir le comportement de la variable de manière différenciée entre le contexte cancéreux et le non cancéreux, nous devons relier les éléments préexistants à la nouvelle variable. Pour ce faire nous formaliserons deux multiplexes régulant la production et la réduction des ROS

dans un contexte cancéreux (donc sur-exprimé). Comme les NOX sont les principales enzymes productrices de ROS nous appellerons le multiplexe activateur **NOX_c**, où **c** indique une activation lorsque nous sommes dans une cellule cancéreuse. Par réciprocity la plus grosse quantité de détoxication se fait à travers la glutathion (GSH), nous appellerons donc le multiplexe réducteurs de ROS **GSH_c**, où **c** indique une inhibition dans un cadre cancéreux. Le graphe de régulation enrichi après insertion de la variable **ROS** et ses régulateurs est donné en figure 10.5.

Construction du multiplexe NOX_c Le multiplexe **NOX_c** doit rassembler dans sa formule les différentes coopérations possibles et nécessaires pour activer la production des ROS. Comme nous l'avons indiqué précédemment, les ROS sont principalement produits *via* la phosphorylation oxydative : **PHOX** ≥ 1 . Pour que la création de ROS se fasse à l'aide de l'enzyme NOX, il faut qu'il y ait suffisamment de *NADPH*. Le *NADPH* est représenté par ratio de *NADPH/NADP⁺* dans le modèle *via* la variable **NADH** : il faut donc que **NADH** ≥ 1 . Enfin, l'augmentation de la quantité de ROS intracellulaire est une conséquence de l'apparition du cancer. Dans notre modèle, le contexte cancéreux étant représenté par les mutations du gène **KRAS**, nous devons l'ajouter à la formule. Au final, le multiplexe **NOX_c** que nous ajoutons au graphe de régulation a pour formule : $((NADH \geq 1 | PHOX \geq 1) \ \& \ KRAS \geq 1$

Construction du multiplexe GSH_c Pour que les ROS soient correctement réduits il faut suffisamment de glutathion disponible dans la cellule. Le glutathion est une protéine composée de trois acides aminés dont les gènes sont sur exprimés dans le cancer du pancréas [Forman et al., 2009]. Pour qu'il y ait détoxication des ROS par le glutathion dans le cancer il peut y avoir création de glutathion *de Novo*. La formation *de Novo* de GSH est possible lorsque l'ATP disponible est présent en suffisamment grande quantité (**ATP** ≥ 1) ou pour répondre à une augmentation du niveau des ROS (**ROS** ≥ 1) [Storz, 2006]. Pour former de la GSH il faut de la glycine, cystéine et de la glutamine, trois acides aminés provenant de l'intérieur ou de l'extérieur de la cellule (représentés par **AA** ou **NCD**). La formule de création du GSH dans la cellule est donc $(AA \geq 1 | NCD \geq 1) \ \& \ (ATP \geq 1 | ROS \geq 1)$. Enfin, le processus de détoxication de la GSH n'est efficace que si sa régénération, dont les étapes sont données figure 10.4 est effective. Son fonctionnement nécessite du *NADPH* disponible dans la cellule : **NADH** ≥ 1 . Finalement, la formule définissant la totalité du multiplexe est donc : $(AA \geq 1 \ | \ NCD \geq 1) \ \& \ (ATP \geq 1 \ | \ ROS \geq 1) \ \& \ NADH \geq 1$

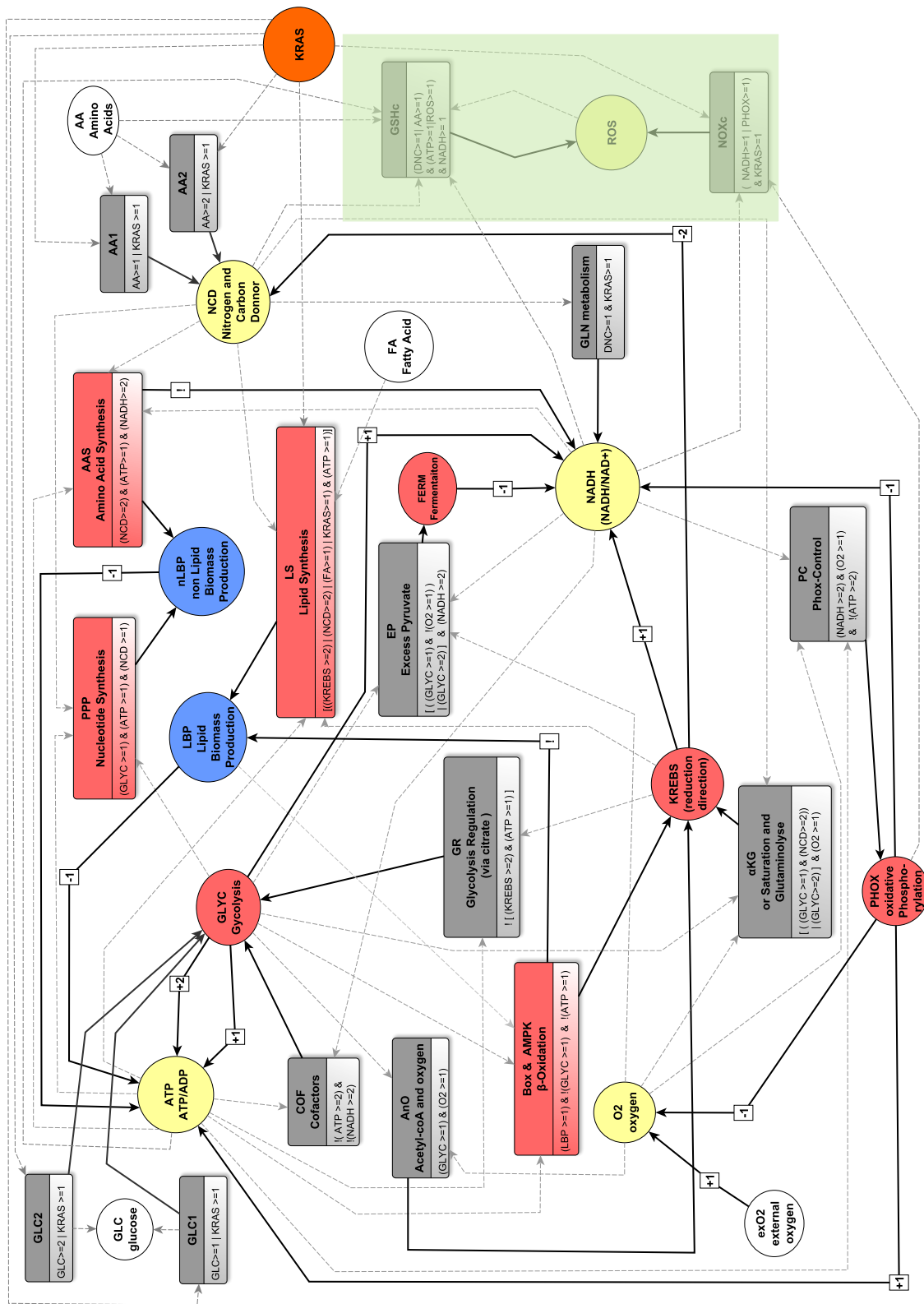


Figure 10.5 – Graphe de régulation après insertion de la variable *ROS* et des multiplexes régulateurs *NOXc* et *GSHc* (encadré vert).

10.3.2.2 Modification cohérente des paramètres de ROS

La variable **ROS** nouvellement ajoutée a deux ressources potentielles et donc 4 paramètres (voir figure : K_{ROS} , $K_{ROS, GSHc}$, $K_{ROS, NOXc}$, $K_{ROS, GSHc NOXc}$). Aucune de ces combinaisons n'est structurellement non applicable. Pour le paramètre K_{ROS} , aucune des ressources n'est présente, ce qui signifie qu'il n'y a pas surproduction des ROS (via NOXc) et qu'en plus il y a une détoxification efficace (via la présence de l'inhibiteur GSHc). Dans ces conditions la quantité de ROS est similaire à celle attendue dans le cadre d'une cellule saine donc $K_{ROS} = 0$. En ce qui concerne les trois autres paramètres ce sont des cas ambiguës on laissera donc *TotemBioNet* déterminer les contraintes de cohérence sur ces paramètres.

10.3.2.3 Modification de la matrice de validation

La présence de la variable **ROS** engendre la création d'une nouvelle colonne dans la matrice de validation.

↪ **1-36** – $FA=0$ ou 1, $exO2=0$ ou 1, $GLC=0$ ou 1 ou 2, $AA=0$ ou 1 ou 2, $KRAS=0$. : L'ancienne matrice éditée dans la section précédente avec les prédictions peut être entièrement réutilisée pour compléter les lignes ↪ 1 – 36 car tous les comportements restent les mêmes, puisque les ROS n'influent pas sur le métabolisme directement. Dans cet environnement, les cellules ne sont pas cancéreuses, ce qui signifie que le niveau des ROS est relativement faible, la variable **ROS** doit donc tendre vers 0.

↪ **37 et 38** – $FA=0$ ou 1, $exO2=0$ ou 1, $GLC=0$ ou 1 ou 2, $AA=0$ ou 1 ou 2, $KRAS=1$. : Lorsque $KRAS=1$ nous sommes dans un contexte cancéreux, ce qui signifie que le taux des ROS doit être supérieur à celui des ROS lorsque l'on est dans un cas non cancéreux : **ROS** ne doit pas tendre vers 0. Pour le reste des comportements, comme il n'y a aucune différence dans la formalisation des autres variables, on va s'appuyer sur la matrice donnée dans l'enrichissement précédent (table 10.5). Le résultat de l'ensemble des modifications est visualisable dans la table 10.6.

Environnements	KRAS	FA	exO2	GLC	AA	ATP (0-2)	O2 (0-1)	GLYC (0-2)	nLBP (0-1)	LBP (0-1)	FERM (0-1)	KREBS (0-2)	PHOX (0-1)	NADH (0-1)	NCD (0-2)	ROS (0-2)
Sans Mutations																
1 à 36	0	0 & 1	0 & 1	0 & 1 & 2	0 & 1 & 2	Identique aux lignes ↪ 1 – 36 de la matrice avant enrichissement donné table 10.5										!td(0)
Mutation KRAS																
38	1		1			Identique aux lignes ↪ 37 et 38 de la matrice avant enrichissement donné table 10.5.										!td(0)

Table 10.6 – Matrice de validation pour le modèle de la régulation du métabolisme avec insertion de la variable **ROS**. Les lignes ↪1 à ↪38 correspondent aux lignes déjà exhibées dans le modèle général de la régulation du métabolisme après l'enrichissement précédent excepté pour la colonne ROS qui constitue de nouvelles spécifications.

10.3.3 Validation par rapport aux connaissances biologiques

TotemBioNet permet de valider 150 paramétrisations pour l'enrichissement proposé. Avec un tel nombre de modèles validés et comme les cycles d'enrichissement ne sont pas terminés, nous devons trier ces modèles et ne conserver que les plus pertinents. En effet, chaque enrichissement est susceptible d'ajouter de nouveaux paramètres. Or, l'ajout de seulement quelques paramètres peut faire exploser le nombre de paramétrisations finales comme le montre la formule donnée chapitre 3.2.2, page 28. Dans ces cas de figure, l'étape de la prédiction devient indispensable pour

discriminer les modèles validés par leurs comportements et ne conserver que le nombre le plus bas possible.

10.3.4 Prédications et retour à la biologie

Parmi les 150 paramétrisations validées, le comportement de **ROS** varie. Cependant pour chacune de prédictions les lignes $\rightsquigarrow 37$ et $\rightsquigarrow 38$ exhibent le même comportement. Comme le seul élément de l'environnement qui change est la quantité d'oxygène, cela indique que la régulation des ROS n'est pas dépendante du niveau d'oxygène disponible pour la cellule, malgré le fait que la production des ROS est en partie due à la phosphorylation oxydative. L'inventaire des différents comportements de la variable **ROS** est le suivant :

- 60 modèles exhibent une oscillation entre 0 et 1 ,
- 60 modèles tendent vers 1,
- 30 modèles tendent vers 2,
- aucun modèle n'oscille entre 1 et 2 ni entre 0 et 2.

Nous décidons de mettre de côté les comportements oscillatoires entre 0 et 1, car une cellule cancéreuse est connue pour avoir un métabolisme général plus élevé qu'une cellule saine. Par défaut, la quantité de ROS produite étant une conséquence du fonctionnement du métabolisme, il paraît peu probable que le niveaux des ROS redescende même ponctuellement à un niveau sain.

Les modèles où **ROS** tend vers 2 sont des modèles que nous décidons de mettre de côté car rien ne nous indique que biologiquement la cellule cancéreuse est capable de survivre à de très hauts niveau de ROS, qui peuvent être considérés comme mortels.

Finalement nous conservons les modèles qui tendent vers 1 pour la variable **ROS**. Cela nous permet de restreindre le nombre de modèles à 60. En regardant ces modèles, $K_{ROS, NOXc} = 1$ et $K_{ROS, GSHc NOXc} \geq 1$. Pour la suite des enrichissements nous conserverons donc les paramètres associés à ce comportement.

A l'aide du premier enrichissement nous avons ajouté un contexte génétique au modèle de la régulation du métabolisme général. Cela a permis de mettre en place un environnement propice pour faire évoluer le contexte de modélisation et la représentation du métabolisme dans le PDAC. Le second enrichissement permet de spécifier (au niveau des cofacteurs) un état cellulaire cancéreux. Enfin le troisième élément, les ROS, est un marqueur connu des cellules du PDAC qui permet de faire le lien entre l'état cellulaire et l'état cancéreux. Après ces trois enrichissements, il reste 60 modèles potentiels qui satisfont l'ensemble des spécifications. L'observation des états de la variable **NADH** et de la variable **ROS** permet d'avoir une double vérification de l'état cancéreux ou non. Certains autres enrichissements auraient pu être effectués en se basant sur d'autres connaissances biologiques. Cependant, l'enrichissement d'un modèle est orienté pour répondre à une problématique en biologie. Dans son état le modèle peut déjà donner des ébauches de réponses sur des régulations potentielles entre les différents éléments du métabolisme. Enfin, et ce sera le sujet du prochain chapitre, les enrichissements proposés pour spécifier le contexte de modélisation au cancer du pancréas sont une base nécessaires et suffisante pour représenter la régulation du métabolisme dans une cellule cancéreuse agressive.

CHAPITRE 11

Enrichissements supplémentaires pour les cellules agressives

Ce chapitre décrit les différents enrichissements utiles pour obtenir un modèle de la régulation du métabolisme d'une cellule cancéreuse agressive dans le cancer du pancréas. A partir du contexte de modélisation donné en fin de chapitre 10, nous décrivons les concepts biologiques et leur formalisation, toujours en suivant la démarche incrémentale du chapitre 4.

11.1 Cancer agressif vs. non-agressif	160
11.1.1 Concepts généraux	160
11.1.2 Concepts biologiques liés à la méthionine	160
11.2 Cycle d'enrichissement pour la méthionine	162
11.2.1 Formalisation du système de régulation	162
11.2.2 Validation de l'enrichissement	165
11.2.3 Prédiction et analyse de l'ajout de la méthionine	165
11.3 Cycle d'enrichissement pour les régulations mettant en cause l'agressivité	166
11.3.1 Formalisation du système de régulation	166
11.3.2 Validation de l'enrichissement	169
11.3.3 Prédiction et analyse des résultats	169

Toujours en s'appuyant sur la méthodologie d'enrichissements successifs décrite chapitre 4, nous proposons ici deux étapes d'enrichissement permettant d'obtenir un premier modèle de la régulation du métabolisme pour une cellule agressive dans le PDAC.

Le but sera d'utiliser ce modèle enrichi pour tester des hypothèses de régulations et ainsi orienter les recherches des biologistes vers certaines voies métaboliques prometteuses. Une première section met en avant les concepts biologiques liés à l'agressivité dans le cancer. Ensuite les sections 11.1.2 et 11.2 de ce chapitre décrivent formellement les différentes étapes des enrichissements. Les enrichissements proposés dans cette section sont contextuels au travail et aux découvertes des biologistes du CRCM de Marseille dans le cadre du projet PAIR-pancréas auquel notre équipe participe.

11.1 Cancer agressif vs. non-agressif

11.1.1 Concepts généraux

On peut schématiser l'évolution du PDAC en deux grandes étapes. La première correspond à l'évolution de la tumeur primaire qui peut se visualiser par l'agrandissement de la tumeur, et la seconde voit certaines cellules migrer pour former une ou plusieurs métastases dans les organes voisins. On considère que le cancer est agressif lorsqu'il y a dissémination des cellules cancéreuses loin de la tumeur principale dans l'organisme. Le passage de l'étape non agressive à l'étape agressive se fait par l'intermédiaire de l'EMT (transition épithélio-mésenchymateuse). L'EMT est un processus de modification de type cellulaire : la cellule épithéliale (considérée comme non agressive) devient mésenchymateuse (agressive). Pendant ce processus les cellules impactées deviennent capable de sortir de l'épithélium du pancréas et d'envahir un organe voisin (dans la majorité des cas, le foie). Les cellules épithéliales et mésenchymateuses sont facilement différenciables, car elles ont des morphologies distinctes, ne possèdent pas les mêmes marqueurs moléculaires, et leur capacité de division cellulaire est différente. Tous ces éléments permettent aux équipes de biologistes de séparer les deux types cellulaires, mais nous avons très peu de connaissances comparatives d'un point de vue métabolique pour différencier ces deux types cellulaires.

L'équipe de Sophie Vasseur au CRCM de Marseille s'intéresse particulièrement à ces cellules agressives et leurs particularités métaboliques dans le but de développer des stratégies pour bloquer l'évolution du PDAC vers l'agressivité. L'équipe s'est premièrement intéressée aux ROS contenus dans la cellule avec des analyses à large spectre. Ils ont découvert que le niveau des ROS dans une cellule cancéreuse agressive était significativement plus bas que dans la cellule cancéreuse non agressive. En se basant sur ce constat, ils ont cherché les différences notables dans les mécanismes de réduction des ROS et ont découvert des dérégulations au niveau du métabolisme de la méthionine.

11.1.2 Concepts biologiques liés à la méthionine

La méthionine est un acide aminé essentiel nécessaire à la croissance et au développement normal des cellules. Chez l'Homme, la méthionine peut être obtenue à partir d'aliments. De plus, elle joue un rôle majeur grâce à son métabolisme, qui alimente une variété de voies métaboliques [Pfeiffer and Morley, 2014]. Le métabolisme de la méthionine peut être divisé en trois parties : la voie de transsulfuration, la voie de la méthionine *de novo* et la voie de sauvetage de la méthionine (dessinées respectivement en rose, violet et bleu dans la figure 11.1) [Parkhitko et al., 2019].

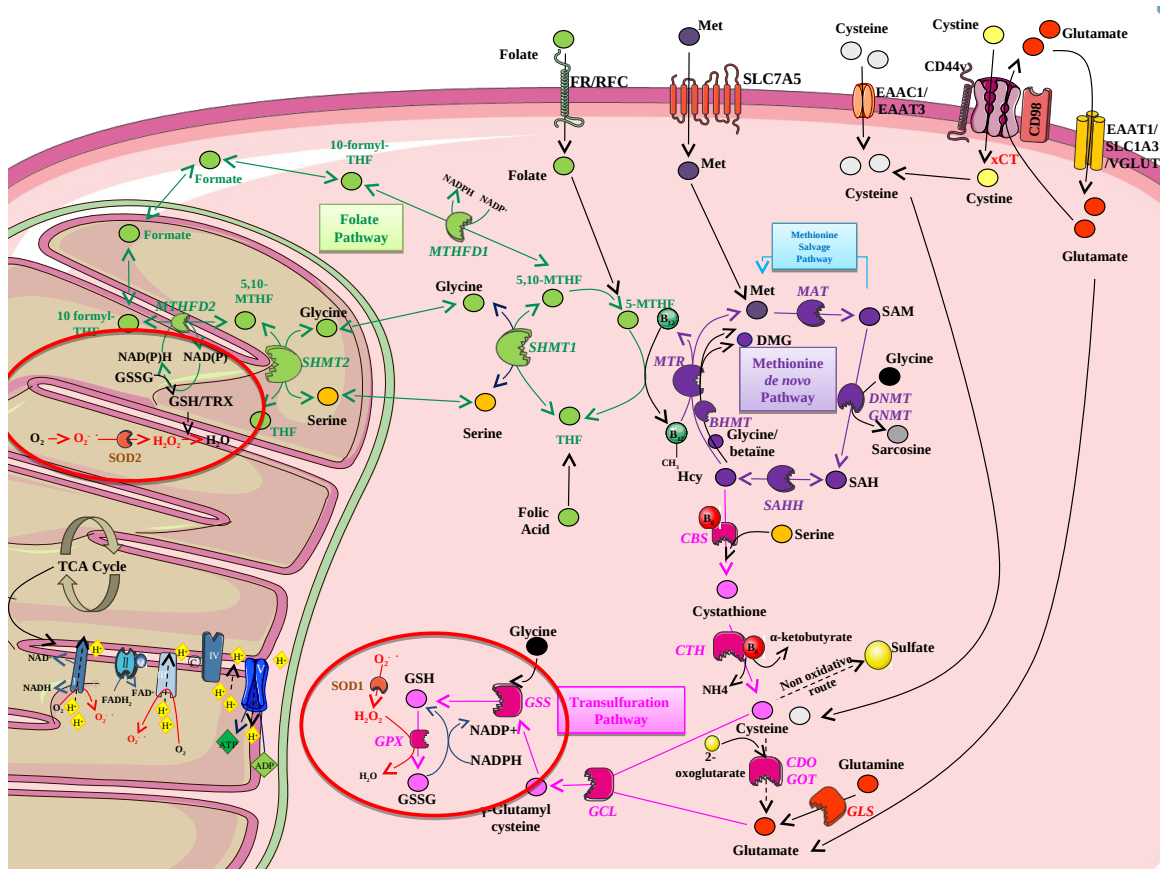


Figure 11.1 – Voies métaboliques liées à la méthionine. En vert le cycle du Folate, en rose la voie de la transsulfuration, en bleu la voie de sauvetage de la méthionine et en violet la voie de la méthionine *de Novo*. Les réactions permettant la réduction des ROS sont entourées en rouge. Support construit par Sophie vasseur.

Différents travaux indiquent que la méthionine et les voies métaboliques liées à cet acide aminé sont importantes pour le développement du cancer [Sullivan et al., 2021]. Ainsi, l'équipe du CRCM a mené des expériences et a découvert que l'ensemble des métabolites impliquées de la voie de la méthionine *de novo* sont beaucoup plus représentées dans la cellule cancéreuse agressive que dans la cellule cancéreuse non agressive dans le cadre particulier du PDAC (résultats des expériences en annexe ??). Cette dérégulation favorise l'entrée vers la voie de la transsulfuration qui permet la création de GSH (voie rose dans la figure 11.1). Cette GSH permet ensuite une détoxification des ROS plus performante de la cellule. Lorsque la voie de la méthionine *de novo* est bloquée, le niveau des ROS remonte et l'agressivité de la cellule diminue. Cela indique que le mécanisme de détoxification *via* la transsulfuration est primordiale pour l'agressivité.

11.2 Cycle d'enrichissement pour la méthionine

La seule piste métabolique que nous ayons pour spécifier le modèle et représenter la cellule cancéreuse concerne l'état des ROS, lui-même dépendant de la méthionine et de ses dérégulations. Le métabolisme de la méthionine devient donc dans ce cadre un point clef qui va nous permettre d'enrichir le modèle. Il convient donc d'enrichir le modèle en ajoutant une nouvelle variable : *MET de novo*.

11.2.1 Formalisation du système de régulation

11.2.1.1 Formalisation du graphe de régulation

On souhaite insérer la variable *MET de novo*, pour représenter la dérégulation de la voie de la méthionine *de novo* dans une cellule cancéreuse agressive. La variable sera booléenne, car soit il y a sur-expression de la voie de la méthionine *de Novo* dans ce cas $MET\ de\ novo = 1$, soit il n'y a pas sur-expression et $MET\ de\ novo = 0$. Par souci de simplicité vis à vis de l'enrichissement, on commencera par représenter la variable comme si c'était une variable d'environnement. On ne prendra donc en compte pour cet enrichissement que les régulations en aval de la variable. Nous ne formaliserons finalement que les régulations qui seront en aval de la variable.

Multiplexe en aval de la variable. La dérégulation de la voie de la méthionine *de novo* ($MET\ de\ novo \geq 1$) provoque une chute du niveau de ROS. Cette chute est due à l'augmentation de la quantité de GSH produite. La régénération du pool de GSH nécessite du NADPH pour que la réduction des ROS soit effective dans la cellule cancéreuse agressive. En plus de la dérégulation de la voie de la méthionine *de novo* un apport suffisant en NADPH ($NADH \geq 1$) est donc nécessaire. On peut donc dessiner une régulation négative de *MET de novo* sur *ROS* par l'intermédiaire du multiplexe **GSHa** (GSHa pour GSH produite dans un état agressif) dont la formule est $!(MET\ de\ novo \geq 1 \ \& \ NADH \geq 1)$. Notons que la négation « ! » indique l'effet inhibiteur de la régulation.

Par ailleurs, rien ne nous indique que la dérégulation de la voie de la méthionine *de novo* ait un impact sur d'autres éléments du métabolisme, il n'y a pas donc lieu de représenter d'autres régulations. En revanche, l'ajout d'une régulation à la variable des ROS nous incite à réévaluer ses niveaux.

Conséquences sur l'abstraction de ROS. La variable *ROS* possède 3 niveaux, 0 étant le niveau sain de ROS, 1 étant le niveau cancéreux et 2 le niveau pour une cellule en apoptose. Sachant que le niveau des ROS d'une cellule agressive est légèrement inférieur à celui de la cellule non agressive, on dédouble le niveau 1.

Conceptuellement, le niveau 0 reste inchangé et le niveau 2 devient niveau 3 sans que le sens soit modifié. Le nouveau niveau 1 correspond à un niveau de ROS trop élevé pour être sain, mais trop peu élevé pour être cancéreux agressif. Enfin, le nouveau niveau 2 sera le niveau le plus élevé de ROS viable (état cancéreux non agressif). *ROS* ne participe qu'à une régulation en aval via le multiplexe **NOXc** au niveau 1. Quel que soit l'état de la cellule (agressif ou non), il y a surproduction de ROS dès que l'on entre dans un état cancéreux. Ce multiplexe qui représente cette surproduction doit pouvoir s'appliquer, que la cellule soit agressive ou non : le niveau indi-

qué dans le graphe de régulation ne doit donc pas être modifié dans le graphe de régulation enrichi.

Remarque – Nous aurions pu nous contenter de modifier l'interprétation des niveaux 1 et 2 pour éviter d'ajouter un niveau supplémentaire, éviter ainsi d'augmenter le nombre de valeurs possibles pour chaque paramètre et ne pas augmenter le nombre de paramétrisations à énumérer. Cependant, le modèle est ici enrichi dans le but de proposer de nouvelles hypothèses de régulations dans le cadre cancéreux agressif sur les différentes voies métaboliques. Prendre en compte ce niveau supplémentaire a donc l'intérêt de laisser au réseau construit un plus large spectre prédictif.

Après l'ensemble de ces enrichissements le graphe de régulation final est donné dans la figure ci-dessous.

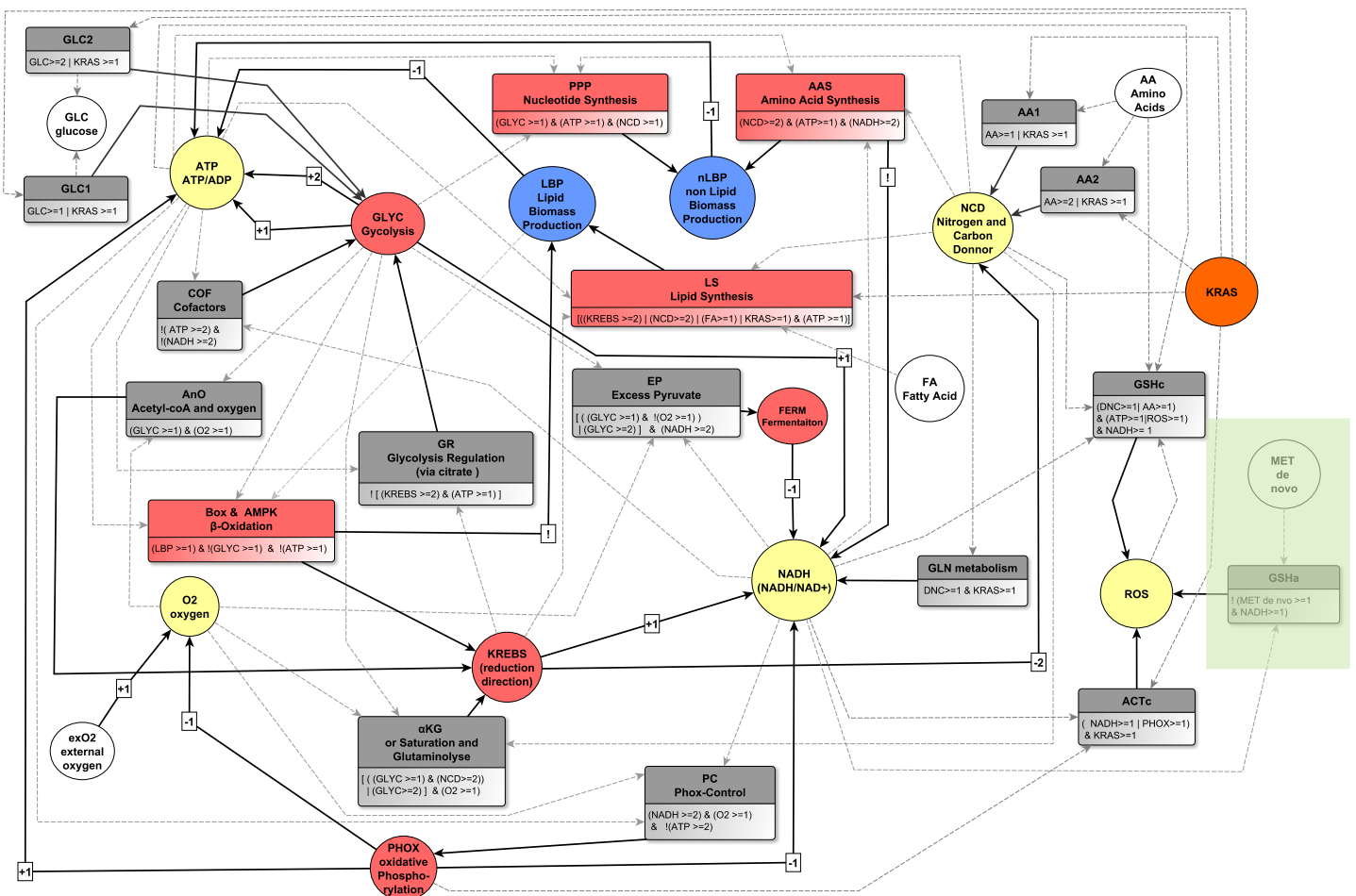


Figure 11.2 – Graphe de régulation enrichi avec la variable *MET de novo* (encadré vert) pour représenter la régulation du métabolisme dans le cadre d'une cellule cancéreuse agressive.

11.2.1.2 Identification des paramètres associés

La variable *MET de novo* est à ce stade une variable d'environnement, il n'y a donc pas de paramètres à identifier. En revanche la modification du nombre de niveaux pour la variable *ROS* nous indique que ces 4 paramètres initiaux identifiés dans la section 10.3.2, page 154 du chapitre précédent sont à réévaluer. De plus, nous ajoutons une ressource à la variable ce qui double son nombre de paramètres (passant à 8).

Dans le contexte de modélisation préexistant nous avons les 4 paramètres suivants : K_{ROS} , $K_{ROS, GSHc}$, $K_{ROS, NOXc}$, $K_{ROS, GSHc NOXc}$. Nous avons ensuite ajouté un inhibiteur (*GSHa*). Ces mêmes paramètres dans le contexte de modélisation enrichi prennent en compte la présence de cet inhibiteur, car la présence d'un inhibiteur correspond à l'absence de la ressource.

La réévaluation des paramètres préexistants. K_{ROS} était déjà égal à 0 *a fortiori* si on ajoute un inhibiteur sans pour autant ajouter d'activateur le niveau des ROS doit être égal à 0. Concernant $K_{ROS, NOXc}$ lorsqu'on a l'activateur cancéreux et les deux inhibiteurs alors on doit avoir un niveau de ROS cancéreux agressif : $K_{ROS, NOXc} = 1$. Enfin, pour les paramètres préexistants ($K_{ROS, GSHc}$ et $K_{ROS, GSHc NOXc}$) les connaissances biologiques ne sont pas suffisantes pour les identifier correctement.

Une fois les paramètres préexistants revus il reste donc à évaluer les 4 paramètres ajoutés par l'enrichissement à identifier pour *ROS* : $K_{ROS, GSHa}$, $K_{ROS, GSHa GSHc}$, $K_{ROS, GSHa NOXc}$, $K_{ROS, GSHa GSHc NOXc}$.

Estimation des nouveaux paramètres Lorsque nous avons l'activateur NOXc et l'inhibiteur GSHc mais pas l'inhibiteur dû à la dérégulation de la méthionine GSHa, le niveau de ROS représenté est un niveau cancéreux non agressif : $K_{ROS, GSHa NOXc} = 2$. Lorsque aucun inhibiteur n'est présent mais que l'activateur est présent, le niveau des *ROS* dans la cellule tend vers son maximum : $K_{ROS, GSHa GSHc NOXc} = 3$. Parmi les 4 nouveaux paramètres seuls 2 ont été identifiés.

En faisant le bilan, parmi les 8 paramètres, dans la catégorie des paramètres préexistants il reste 2 paramètres non identifiés ($K_{ROS, GSHc}$ et $K_{ROS, GSHc NOXc}$) et le même nombre ($K_{ROS, GSHa}$ et $K_{ROS, GSHa GSHc}$) dans la catégorie des paramètres ajoutés par l'enrichissement.

11.2.1.3 Modification de la matrice de validation

MET de novo est une variable d'environnement booléenne, elle double donc le nombre de lignes de la matrice (donnée table 11.1). On passe donc de 38 lignes (notées $\rightsquigarrow 1-38$) à 76 lignes. $\rightsquigarrow 1-36$ – $FA=0$ ou 1, $exO2=0$ ou 1, $GLC=0$ ou 1 ou 2, $AA=0$ ou 1 ou 2, $KRAS=0$ et $MET de novo=0$: Les comportements des variables dont le sens abstrait n'a pas été réévalué peuvent être conservés dans les environnements où $MET de novo = 0$. Seule la colonne qui décrit le comportement de *ROS* doit être réévaluée. Comme nous sommes dans des cas non cancéreux le niveau des ROS doit être considéré comme sain : *ROS* doit donc tendre vers 0.

$\rightsquigarrow 37$ et 38 – $FA=0$ ou 1, $exO2=0$ ou 1, $GLC=0$ ou 1 ou 2, $AA=0$ ou 1 ou 2, $KRAS=1$ et $MET de novo=0$: Sans la dérégulation de la méthionine, pour toutes les lignes où $KRAS=1$ il y a mutation génétique et la cellule est cancéreuse non agressive : le niveau de ROS est donc supposé élevé : *ROS* doit donc tendre vers 2. L'environnement présent a déjà été rencontré dans le cadre de l'enrichissement pour obtenir un modèle de la régulation du métabolisme dans le cancer du pancréas, voir chapitre

Environnements	MET de novo	KRAS	FA	exO2	GLC	AA	ATP (0-2)	O2 (0-1)	GLYC (0-2)	nLBP (0-1)	LBP (0-1)	FERM (0-1)	KREBS (0-2)	PHOX (0-1)	NADH (0-1)	NCD (0-2)	ROS (0-2)
Sans Mutations																	
1 à 36	0	0	0 & 1	0 & 1	0 & 1 & 2	0 & 1 & 2	Identique matrice modèle donné en table 10.6										
Mutation KRAS																	
37 et 38	0	1		1			Identique matrice modèle donné en table 10.6										td(2)
Mutation MET de novo sans mutation KRAS																	
39 à 74	1	0	0 & 1	0 & 1	0 & 1 & 2	0 & 1 & 2	Non estimé car conditions non retrouvées biologiquement										
Mutation MET de novo avec mutation KRAS																	
75 et 76	1	1		0 & 1			Non estimé car phénotypes non identifiés										!td(0) & !td(2)

Table 11.1 – Matrice de validation pour le modèle de la régulation du métabolisme avec insertion de la variable *Met de novo*. Les lignes $\rightsquigarrow 1$ à $\rightsquigarrow 38$ correspondent aux lignes déjà spécifiées dans le modèle général de la régulation du métabolisme après l’enrichissement précédent excepté pour la colonne ROS qui requiert de nouvelles spécifications.

10. Dans le dernier cycle d’enrichissement qui vise à ajouter les *ROS* nous savons que les comportements cancéreux pour les variables *ATP, O2, GLYC, LBP, nLBP, KREBS, FERM* et *PHOX, NADH, NCD* sont identiques aux comportements dans le cadre de la glycolyse en aérobie, ce qui nous a permis de reporter les comportements de ces conditions.

\rightsquigarrow **39-74** – *FA*=0 ou 1, *exO2*=0 ou 1, *GLC*=0 ou 1 ou 2, *AA*=0 ou 1 ou 2, *KRAS*=0 et *MET de novo*=1 : Quand la méthionine est dérégulée, mais pas *KRAS* nous sommes dans un contexte non défini supposé pathologique non retrouvé biologiquement. Aucune estimation des comportements ne peut être faite. Même si ces environnements existent théoriquement, leur contenu ne doit pas écarter de modèles potentiels. L’ensemble des cases est donc vide.

\rightsquigarrow **75 et 76** – *FA*=0 ou 1, *exO2*=0 ou 1, *GLC*=0 ou 1 ou 2, *AA*=0 ou 1 ou 2, *KRAS*=1 et *MET de novo*=1 : Quand la méthionine est dérégulée comme *KRAS* nous sommes dans un contexte cancéreux agressif. Nous savons que le niveau des ROS n’est pas sain, !td(0), mais il ne doit pas être aussi élevé que dans le cadre de la cellule cancéreuse non agressive donné par les lignes $\rightsquigarrow 37$ & $\rightsquigarrow 38$, on indique donc pour les ROS « !td(0) & !td(2) ».

11.2.2 Validation de l’enrichissement

L’enrichissement proposé dans cette section possède alors 30 paramétrisations valides avec *TotemBioNet*. Si l’on regarde attentivement ces dernières leur nombre est uniquement du aux paramètres des variables *ATP* et *NADH* non identifiés¹ depuis la mise en place du premier modèle de la régulation du métabolisme chapitre 9). Les 30 paramétrisations exhibent toutes les mêmes valeurs (respectivement 0, 0 et 1) pour les paramètres $K_{ROS, GSHc}$, $K_{ROS, GSHa}$ $GSHc$ et $K_{ROS, GSHc}$ $NOXC$. Pour la suite des enrichissements nous utiliserons donc ces valeurs pour les paramètres estimés formellement. Ces valeurs constituent aussi des prédictions sur le comportement local des réactions.

11.2.3 Prédiction et analyse de l’ajout de la méthionine

Les 30 paramétrisations validées, prédisent toutes que lorsque l’on est dans des conditions cancéreuses agressives *ROS* tend vers 1. Si l’on regarde les prédictions sur les comportements des variables liées uniquement au métabolisme, on ne constate aucune autre différence comportementale entre l’état cancéreux non agressif et agressif. Sur ce point, pour l’instant nous n’avons pas fait d’hypothèse de régulation directement sur les éléments du métabolisme, il paraît raisonnable

1. $K_{ATP, LBP PHOX}$, $K_{ATP, GLYC1 LBP}$, $K_{ATP, GLYC1 GLYC2 LBP}$, $K_{ATP, GLYC1 LBP PHOX}$, $K_{ATP, GLYC1 GLYC2 LBP PHOX}$, $K_{NADH, FERM GLYC PHOX}$, $K_{NADH, GLNm FERM GLYC PHOX}$, $K_{NADH, FERM GLYC KREBS}$, $K_{NADH, GLNm FERM GLYC KREBS}$

que les comportements ne soient pas différenciés. Si l'on se réfère aux connaissances biologiques sur les différences sur le métabolisme général centré sur le traitement du glucose, nous n'avons pas de différences notables entre ces deux types cellulaires, les prédictions sont donc correctes vis à vis des connaissances biologiques actuelles.

11.3 Cycle d'enrichissement pour les régulations mettant en cause l'agressivité

11.3.1 Formalisation du système de régulation

En laboratoire, l'induction de l'EMT se fait soit par l'ajout d'un facteur de croissance appelé $TGF-\beta$ soit par la réduction de la quantité d'oxygène pour la cellule [Dardare et al., 2020, Salnikov et al., 2012]. Pour se rapprocher du système réel étudié et mener différents test de simulation sur le modèle, nous allons ajouter au modèle cette notion d'induction d'EMT.

11.3.1.1 Modification du graphe de régulation

Ajout du multiplexe EMT Pour formaliser l'induction de l'EMT dans le modèle nous avons besoin d'ajouter un multiplexe qui représente cette régulation : nous l'appellerons **EMT**. Comme nous indique le paragraphe précédent, l'EMT est déclenchée lorsque la cellule cancéreuse ($KRAS \geq 1$) est en hypoxie ($InOA=0$) ou lorsque du $TGF-\beta$ est injecté dans le milieu. L'oxygène est déjà présent mais en revanche aucun des composants du modèle ne peut être lié au $TGF-\beta$: on doit donc ajouter une nouvelle variable pour le représenter que l'on appelle $TGF-\beta$. Cette variable est booléenne.

Avec l'ajout de la variable $TGF-\beta$ la formule du multiplexe **EMT** peut être entièrement formalisée par : $!InOA \geq 1 \mid TGF-\beta \geq 1$ en accord avec les connaissances sus-mentionnées. De plus, comme la voie de la méthionine *de novo* est modifiée dans le cancer agressif, on peut en déduire que l'EMT régule (éventuellement indirectement) cette voie. On peut donc modéliser *via* un arc que le multiplexe **EMT** active *MET de novo*.

Ajout de la variable $TGF-\beta$ Revenons maintenant à la variable $TGF-\beta$. Cette dernière représente une molécule ajoutée en laboratoire pour induire la transition EMT : c'est donc une variable d'environnement. En conséquence il n'y a aucune régulation entrante à définir pour cette variable. De plus, nous avons déjà établi comment la variable était impliquée dans le multiplexe **EMT**, et aucune information biologique ne nous indique qu'il existe d'autres régulations majeures sur les différents éléments du métabolisme représenté.

Nous avons donc terminé l'enrichissement du graphe de régulation : le résultat est donné en figure 11.3.

11.3.1.2 Modification cohérente des paramètres

Avec les enrichissements du graphe de régulation, la variable *MET de novo* qui était une variable d'environnement gagne un prédécesseur (**EMT**). Nous avons donc deux paramètres à estimer : $K_{METde novo}$ et $K_{METde novo, EMT}$. Lorsque l'EMT n'est pas déclenchée, le paramètre $K_{METde novo}$ s'applique. Sans EMT la voie de la méthionine *de novo* n'est pas suractivée et ne

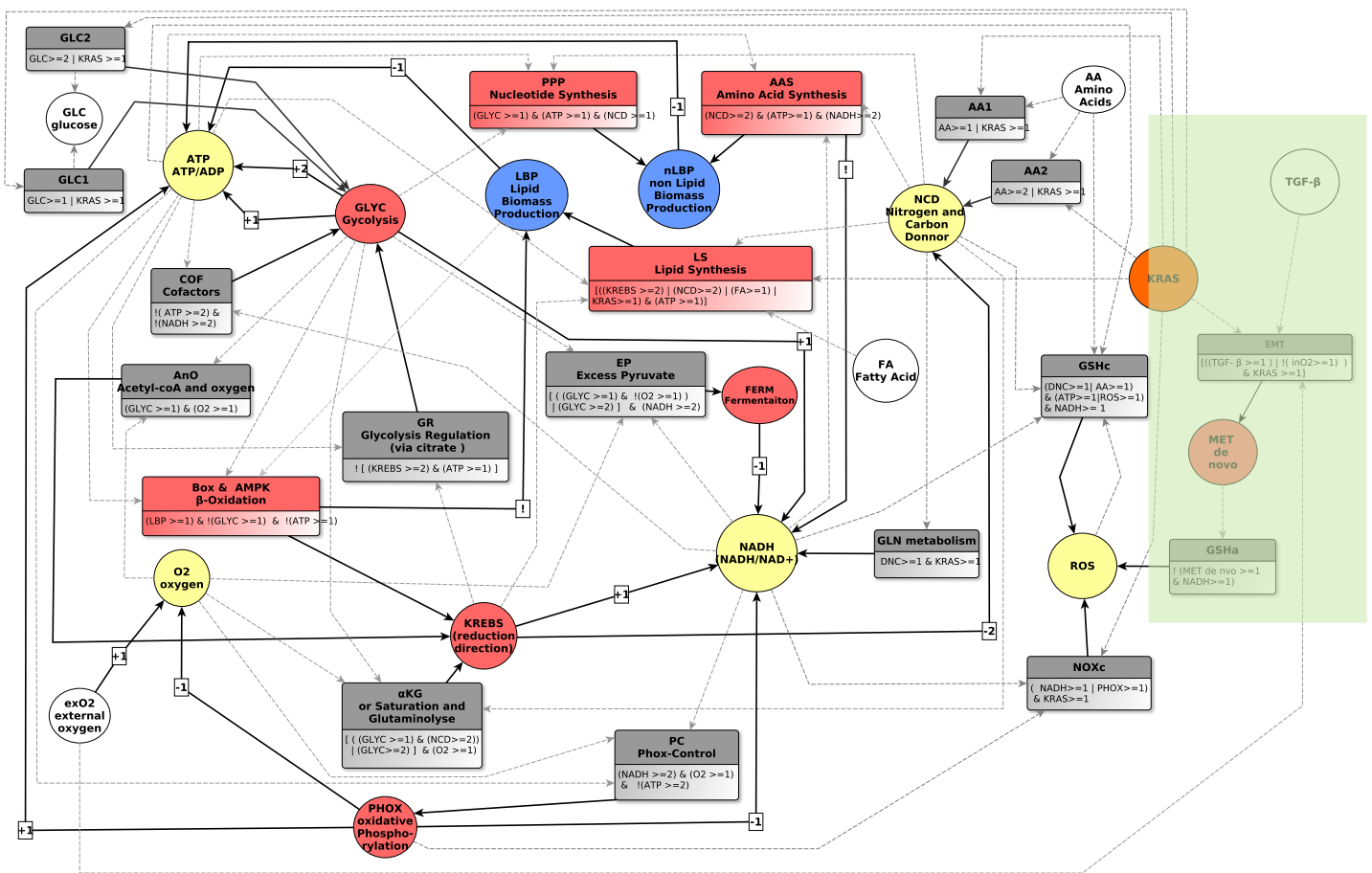


Figure 11.3 – Graphe de régulation après enrichissement avec *TGF-β*.

joue pas son rôle d'inhibiteur sur les ROS $K_{MET\text{denovo}} = 0$. A l'inverse avec EMT la voie de la méthionine *de novo* est suractivée et joue son rôle d'inhibiteur sur les ROS : $K_{MET\text{denovo},\text{EMT}} = 1$.

Rappelons que nous avons établi lors de l'enrichissement précédent que $K_{ROS,\text{GSHc}} = 0$, $K_{ROS,\text{GSHa GSHc}} = 0$ et $K_{ROS,\text{GSHc NOXc}} = 1$.

11.3.1.3 Modification de la matrice de validation

$TGF-\beta$ est une variable d'environnement booléenne, théoriquement elle devrait doubler le nombre de lignes de la matrice précédente. Cependant, la variable *MET de novo* est désormais régulée et ne fait plus partie des variables d'environnement : le nombre de lignes de la matrice enrichie (donnée table 11.1) n'est donc pas modifié. La colonne décrivant les comportements de *MET de novo* est maintenant dans la partie phénotype puisque c'est une variable interne dans le graphe de régulation.

Table 11.2 – Matrice de validation pour le modèle de la régulation du métabolisme avec insertion de la variable d'environnement $TGF-\beta$. Les lignes $\rightsquigarrow 1$ à $\rightsquigarrow 38$ correspondent aux lignes déjà spécifiées dans le modèle général de la régulation du métabolisme après l'enrichissement précédent, excepté pour les colonnes *MET de novo* et *ROS* qui constituent de nouvelles spécifications.

\rightsquigarrow **1-36** – $FA=0$ ou 1, $exO2=0$ ou 1, $GLC=0$ ou 1 ou 2, $AA=0$ ou 1 ou 2, $KRAS=0$ et $TGF-\beta=0$: Les comportements des variables dont le sens abstrait n'a pas été modifié peuvent être conservés dans les environnements où $TGF-\beta = 0$ et $KRAS = 0$. L'ensemble des comportements validés dans l'enrichissement précédent peuvent donc être conservés.

\rightsquigarrow **37** – $FA=0$ ou 1, $exO2=0$ ou 1, $GLC=0$ ou 1 ou 2, $AA=0$ ou 1 ou 2, $KRAS=1$ et $TGF-\beta=0$: Sans *TGF-beta*, sans oxygène ($exO2=0$) et avec la mutation KRAS ($KRAS=1$), la cellule est cancéreuse agressive. Nous savons que le niveau des ROS n'est pas sain, $!td(0)$, mais il ne doit pas être aussi élevé que dans le cadre de la cellule cancéreuse non agressive (qui tend vers 2), on indique donc que les ROS « $!td(2)$ ». Concernant les comportements des autres variables (*ATP, O2, GLYC, LBP, nLBP, KREBS, FERM, PHOX, NADH* et *NCD*) rien ne nous indique que l'on a les mêmes comportements dans un état cancéreux simple et dans un état cancéreux agressif. En fait, tous leurs comportements ne font pas consensus donc laissés vides.

\rightsquigarrow **38** – $FA=0$ ou 1, $exO2=1$, $GLC=0$ ou 1 ou 2, $AA=0$ ou 1 ou 2, $KRAS=1$ et $TGF-\beta=0$: Sans *TGF-beta*, avec oxygène et la mutation KRAS, la cellule est cancéreuse non agressive : *ROS* doit donc tendre vers 2. L'environnement présent à déjà été rencontré dans le cadre de l'enrichissement pour obtenir un modèle de la régulation du métabolisme dans le cancer du pancréas (voir chapitre 10), puis dans ce même chapitre lors de l'ajout de la variable *MET de novo*. Les comportements pour les variables *ATP, O2, GLYC, LBP, nLBP, KREBS, FERM, PHOX, NADH* et *NCD* sont identiques aux comportement dans le cadre de la glycolyse en aérobie ce qui nous a permis de reporter les comportements de ces conditions.

\rightsquigarrow **39-74** – $FA=0$ ou 1, $exO2=0$ ou 1, $GLC=0$ ou 1 ou 2, $AA=0$ ou 1 ou 2, $KRAS=0$ et $MET\text{ de novo}=1$: Dans toutes les conditions où $TGF-\beta=1$ et $KRAS=0$ nous ne sommes pas dans un cadre cancéreux. En conséquence *ROS* tend vers 0. Dans le cadre de notre étude, nous n'avons pas de connaissances sur l'impact du $TGF-\beta$ sur le métabolisme dans les cellules non cancéreuses. Pour l'ensemble des autres éléments, on ne définit aucun comportement consensus (toutes les autres cases sont vides).

\rightsquigarrow **75 & 76** – $FA=0$ ou 1, $exO2=0$ ou 1, $GLC=0$ ou 1 ou 2, $AA=0$ ou 1 ou 2, $KRAS=1$ et $TGF-\beta=1$: Quand *KRAS* est muté et que nous avons à disposition du $TGF-\beta$ nous sommes dans un contexte

cancéreux agressif. Nous savons que le niveau des ROS est similaire à \rightsquigarrow 37 et la méthionine *de novo* dérégulée (« !td(0) »). Concernant les autres variables, nous n’estimons aucun comportement pour les mêmes raisons que pour la ligne \rightsquigarrow 37.

11.3.2 Validation de l’enrichissement

Lors de cet enrichissement tous les nouveaux paramètres ont été identifiés manuellement. Nous aurons donc au maximum 30 paramétrisations. De plus, nous réutilisons les valeurs des paramètres $K_{ROS, GSHc} = 0$, $K_{ROS, GSHa GSHc} = 0$ et $K_{ROS, GSHc NOxc} = 1$ déduits de l’enrichissement précédent (voir section 11.2.2). La validation de cette étape permet de fixer le dernier paramètre restant de la variable **ROS** à 0 puisque toutes les paramétrisations valides ont la même valeur pour ce paramètre $K_{ROS, GSHa}$.

11.3.3 Prédiction et analyse des résultats

En analysant les résultats de *TotemBioNet*, on constate que pour tous les modèles valides, le comportement des ROS est identique. Dans un contexte agressif, le niveau des ROS est plus bas que dans un contexte non agressif, confirmant que le modèle représente bien le système biologique. De plus, nous n’avons pas indiqué de comportement pour les autres variables du métabolisme. Les comportements prédits dans le cadre du cancer agressif sont identiques aux comportements dans le cadre du cancer non agressif. Ceci est tout à fait raisonnable car nous n’avons pas lié les marqueurs de l’agressivité et les éléments internes du métabolisme du modèle.

Après l’ensemble des enrichissements vus dans ce chapitre, le modèle est capable de différencier une cellule cancéreuse agressive d’une cellule cancéreuse non agressive. Cependant, nous n’avons mis en place aucune hypothèse de régulation dans ces différentes conditions. Le modèle construit est un prototype de la régulation du métabolisme dans le cancer du pancréas. Les phases d’enrichissement de ce chapitre nous ont cependant permis de prédire l’évolution locale des régulations au niveau des ROS *via* l’identification automatique des paramètres de cette variable.

CHAPITRE 12

Conclusion et Perspectives

Les travaux présentés dans ce manuscrit sont de plusieurs natures avec, d'une part, une réflexion méthodologique sur la démarche de construction de modèles de systèmes biologiques, et d'autre part une étude en profondeur d'un système biologique qui n'est pas totalement compris (le modèle que nous concevons, accompagnant une recherche en cours en biologie). Ainsi, on peut résumer les apports de nos travaux autour de deux axes : la réflexion sur la démarche de modélisation et le travail sur la modélisation de la régulation du métabolisme, dans un contexte cancéreux, agressif ou non.

12.1 Les acquis de cette thèse

12.1.1 Les apports méthodologiques

Le développement de modèles *ex nihilo* devient de plus en plus complexe. En premier lieu, lorsque les modèles grandissent, le nombre de paramètres dynamiques explose. Les techniques formelles pour l'identification de la totalité de ces paramètres ne sont pas, encore aujourd'hui, praticables en force brute sur des modèles aussi grands que la régulation du métabolisme cellulaire. De plus, même si les connaissances des biologistes et biochimistes permettent de réduire l'explosion combinatoire du nombre de paramètres, cela fait appel à des expériences de pensée dont la mise en place soignée est très longue. Nous sommes donc entrés dans une période où nous avons besoin de développer de nouvelles méthodes pour concevoir des modèles de plus en plus grands et complexes.

En majorité, les grandes fonctions biologiques que l'on cherche à modéliser sont communes à différents organismes vivants (cycle cellulaire, métabolisme...). En se reposant sur des modèles généraux de ces différents systèmes biologiques, on a proposé dans cette thèse (chapitre 4) l'adaptation à notre problématique d'un modèle existant. Pour y parvenir nous avons systématisé de petits enrichissements incrémentaux en suivant une nouvelle méthodologie, que nous avons conçue. [Gibart et al., 2021a]. Cette méthodologie s'inspire du cycle de vie en spirale utilisé en Génie Logiciel. Cette nouvelle méthode repose sur l'existence de modèles initiaux de taille moyenne, qu'il faut concevoir *ex nihilo* en utilisant une méthode de développement apparentée au cycle de développement en « V ».

L'ajout d'un élément dans le modèle est susceptible de remettre en cause tous les comportements (propriétés) préalablement validés. Cependant, l'apport de ces éléments de manière incrémentale permet de cibler plus précisément les risques de régression, en contrôlant finement

les ajouts potentiellement responsables de la perte de propriétés attendues du modèle. De plus, lorsque régression il y a, l'incrément ajouté étant de taille relativement faible, il est plus facile de réviser la formalisation en se fondant uniquement sur des connaissances biologiques.

Nous avons passé en revue et formalisé chaque étape de cette nouvelle méthodologie. La notion de contexte de modélisation offre un inventaire rigoureux des éléments à gérer à chaque instant pour la conception du modèle. Concernant l'étape de modification du contexte de modélisation nous avons proposé une méthodologie rigoureuse de formalisation d'enrichissements qui reprend les connaissances préalablement formalisées et les adapte au modèle enrichi (chapitre 8). Nous avons aussi défini une stratégie de calcul efficace pour le processus de validation du contexte de modélisation nouvellement enrichi (chapitre 7). Cette méthode s'appuie sur une représentation matricielle des phénotypes attendus, une fois le régime stationnaire atteint, pour chaque marqueur biologique, et ce, en fonction de l'environnement [Gibart et al., 2021b]. Dans ce but, nous nous sommes appuyés sur une approche "diviser pour régner" afin de réduire l'explosion combinatoire causée par l'énumération des paramétrisations. Pour ce faire nous avons exploité un nouveau type de variable : les variables d'environnements. L'approche "diviser pour régner" a ensuite été automatisée *via* le développement d'un module de *TotemBioNet* appelé *GreenBioNet* [Gibart et al., 2022].

Ces améliorations méthodologiques ont ensuite été mises en oeuvre à grande échelle avec le cas d'étude concret qu'est la régulation du métabolisme d'un adénocarcinome pancréatique.

12.1.2 Les apports de modélisation : métabolisme cancéreux du pancréas

Dans le cadre de l'application de la méthodologie, nous avons d'abord repassé au crible le modèle préexistant de la régulation générale du métabolisme (premier modèle *ex nihilo* avec méthode en « V »), ceci amenant à la publication d'un article de journal [Gibart et al., 2021c] en collaboration avec les premiers créateurs du modèle.

Dans un second temps, en collaboration avec l'équipe du docteur Sophie Vasseur, nous avons défini un certain nombre de critères que le modèle de la régulation du métabolisme dans le cancer du pancréas devait avoir. Notre premier objectif a été de discriminer les phénotypes de cellules cancéreuses de celui des cellules saines, et cela dans le cadre d'un modèle commun. Plusieurs éléments sont apparus comme indispensables à ajouter au modèle, comme la mutation du gène KRAS, ou encore les différences de niveau des ROS cellulaires. Nous avons donc procédé à une série de trois étapes d'enrichissements pour ajouter ces éléments.

Enfin, pour aller plus loin et pouvoir différencier différents stades de la maladie dans le modèle, nous nous sommes appuyés sur les dernières découvertes de nos partenaires biologistes qui indiquent que le taux de ROS dans une cellule agressive (EMT) est significativement plus bas que dans une cellule cancéreuse non agressive (épithéliale). Ceci s'explique biologiquement par la dérégulation de la voie de la méthionine *de Novo* qui permet une détoxification plus efficace (réduction des ROS). Notre modèle rend compte de ces observations.

12.1.3 Perspectives d'enrichissements et de prédictions

Dans le cadre du modèle de régulation du métabolisme dans le cancer du pancréas, nous avons travaillé sur les premiers enrichissements en nous basant sur des connaissances consensuelles

fiables, données par les biologistes dans le laboratoire. Cependant, au sens strict, si l'on considère que l'enrichissement n'est pas complet ou qu'une nouvelle connaissance apparaît, il est possible d'ajouter de nouveaux éléments. Le modèle que nous avons conçu est bien adapté pour permettre ces enrichissements. En ce sens, nous pourrions faire une liste exhaustive de toutes les pistes d'amélioration possibles, mais cela dépend de l'évolution de la problématique et des besoins des biologistes pour leur apporter des modèles pouvant leur fournir un cadre de raisonnement adapté à leur question du moment. Autrement dit, l'enrichissement du modèle avec de nouveaux éléments doit être motivé par de nouvelles questions biologiques et non pour ajouter des détails concrets. Pour chaque enrichissement choisi, il est important de rester au même niveau d'abstraction et ne pas confondre stratégie d'enrichissement et modélisation multi-échelle. En effet, l'ajout d'un détail qui ne correspond pas au niveau d'abstraction pourrait déséquilibrer le modèle, rendant hétérogènes les formules *CTL* ou de la logique de HOARE, rendant les conclusions du modèle non utilisables en pratique.

Quelques pistes pour l'amélioration du modèle paraissent néanmoins très intéressantes. Premièrement, les connaissances sur le développement morphologique et le cycle cellulaire d'une cellule cancéreuse sont très détaillées. Sachant que la régulation du métabolisme y est liée, il serait pertinent de coupler notre modèle avec le cycle cellulaire pour mieux identifier les différents stades évolutifs du cancer du pancréas et pour représenter une cellule en processus de mort cellulaire (nécrose, apoptose...). Ce couplage permettrait d'effectuer des expériences *in silico* (expériences de KO ou de sur-expression par exemple) visant à mener à la mort par la dérégulation du métabolisme de la cellule cancéreuse. La seconde piste qui mériterait d'être approfondie concerne directement l'équipe du CRCM. En effet, dans le projet, certains membres de l'équipe s'intéressent aux différents lipides dans l'évolution du cancer ainsi qu'à la régulation des voies métaboliques qui y sont associés. Or, le modèle aujourd'hui n'est pas assez détaillé pour l'étude comparative des voies métaboliques spécifiquement liées aux lipides.

Le modèle proposé chapitre 11, nous a d'ores et déjà permis effectuer des tests de régulations, puisque l'on est capable de discriminer l'état de la cellule (cancéreuse ou non). En étudiant le graphe de régulation nous avons constaté que les enrichissements effectués ne sont que partiellement connectés au reste du graphe de régulation initial. L'expertise du biologiste a ensuite permis de mettre en avant que la dérégulation de la voie de la méthionine *de Novo* exerçait probablement une régulation sur d'autres voies métaboliques telles que la synthèse d'acides aminés pour n'en citer qu'une seule. Nous avons donc commencé l'exploitation de notre modèle en effectuant différents tests de régulation associés aux éléments précités. Chacun des tests consiste à ajouter une ou plusieurs régulations fonctionnelles dans le contexte de modélisation sans modifier les autres composants (matrice de phénotypes et paramètres) puis à utiliser *TotemBioNet* pour chercher d'éventuels régressions par rapport aux comportements spécifiés. Les premiers tests montrent qu'il pourrait y avoir une régulation négative de la voie de la méthionine *de Novo* sur différents éléments du métabolisme : une étude plus approfondie de ces tests permettrait de valider des hypothèses sur ces régulations.

La méthodologie développée a permis l'enrichissement d'un modèle de grande taille. Cependant nous avons remarqué que la *prédiction* est une étape limitante car les outils utilisés ne sont pas particulièrement adaptés pour y répondre clairement et de manière concise.

12.2 Aides à la prédiction

La première étape de la méthodologie à reconsidérer est l'étape de prédiction. En effet même si cette étape n'est développée dans aucun chapitre de la thèse, nous y avons déjà travaillé et 3 méthodes au moins sont envisageables pour des prédictions lisibles. Les deux premières ont déjà été utilisées lors de la thèse pour les différents enrichissements décrits aux chapitres 10 et 11. La troisième approche consiste en une méthode que nous pensons prometteuse pour laquelle il reste à faire une preuve de concept.

12.2.1 Prédiction naïve

Dans l'approche naïve nous avons commencé par reconstruire, pour chaque paramétrisation validée, une matrice dite de "Prédiction", puis nous comparions les différents résultats. Cette prédiction naïve repose sur l'utilisation de *TotemBioNet*. A partir des formules *Fair-Path CTL* à tester pour une paramétrisation, l'outil est capable d'indiquer celles qui ne sont pas satisfaites (flèche bleue et flèches noires de la figure 12.1). Avec les patterns décrits dans le chapitre 5 nous pouvons avoir une prédiction des comportements asymptotiques pour toutes les variables du système. L'avantage principal de cette technique est d'obtenir, pour un enrichissement et pour chaque paramétrisation validée, une matrice de comportements prédits, compréhensible par un biologiste avisé. Cette matrice devient donc le support d'un dialogue avec l'expert biologiste. A l'inverse, lorsque le nombre de paramétrisations validées est trop important (> 20 environ), l'analyse de ces matrices devient trop compliquée à faire « à la main ».

12.2.2 Prédiction avec hypothèses

Pour palier cette limitation humaine, une seconde solution a été utilisée pour travailler sur les enrichissements qui validaient « trop » de modèles. L'idée est d'ajouter deux étapes complémentaires à la prédiction naïve des comportements pour réduire le nombre de contextes de modélisation instanciés que l'on aura à interpréter (figure 12.1).

La première étape consiste à ajouter dans la matrice des phénotypes validés, des comportements "*hypothétiques*" (petit 1 de la figure 12.1) pour constituer une matrice dite *hypothétique*. Les comportements que l'on appelle hypothétiques sont ceux que l'on va supposer, mais qui n'ont pas été totalement vérifiés expérimentalement (ils ne font pas consensus). On ajoute ensuite cette matrice hypothétique aux éléments du contexte de modélisation validé (paramétrisations et graphe de régulation) pour obtenir un contexte de modélisation hypothétique (rectangle en haut à droite de la figure 12.1). La seconde étape réside dans l'utilisation de *TotemBioNet* pour obtenir un contexte de modélisation valide vis à vis des comportements hypothétiques (petit 2 de la figure 12.1). Enfin, la dernière étape est similaire à la première méthode : elle consiste à créer puis analyser avec l'aide des biologistes les matrices de prédictions restantes. Cependant ici, les matrices de prédiction obtenues sont contraintes par les comportements hypothétiques, elles sont donc moins nombreuses.

L'avantage principal de la prédiction avec hypothèse est la réduction du nombre de matrices de prédictions à étudier, ce qui facilite l'analyse des prédictions. Cependant, cela repose sur l'utilisation *a priori* d'hypothèses biologiques qui peuvent s'avérer fausses.

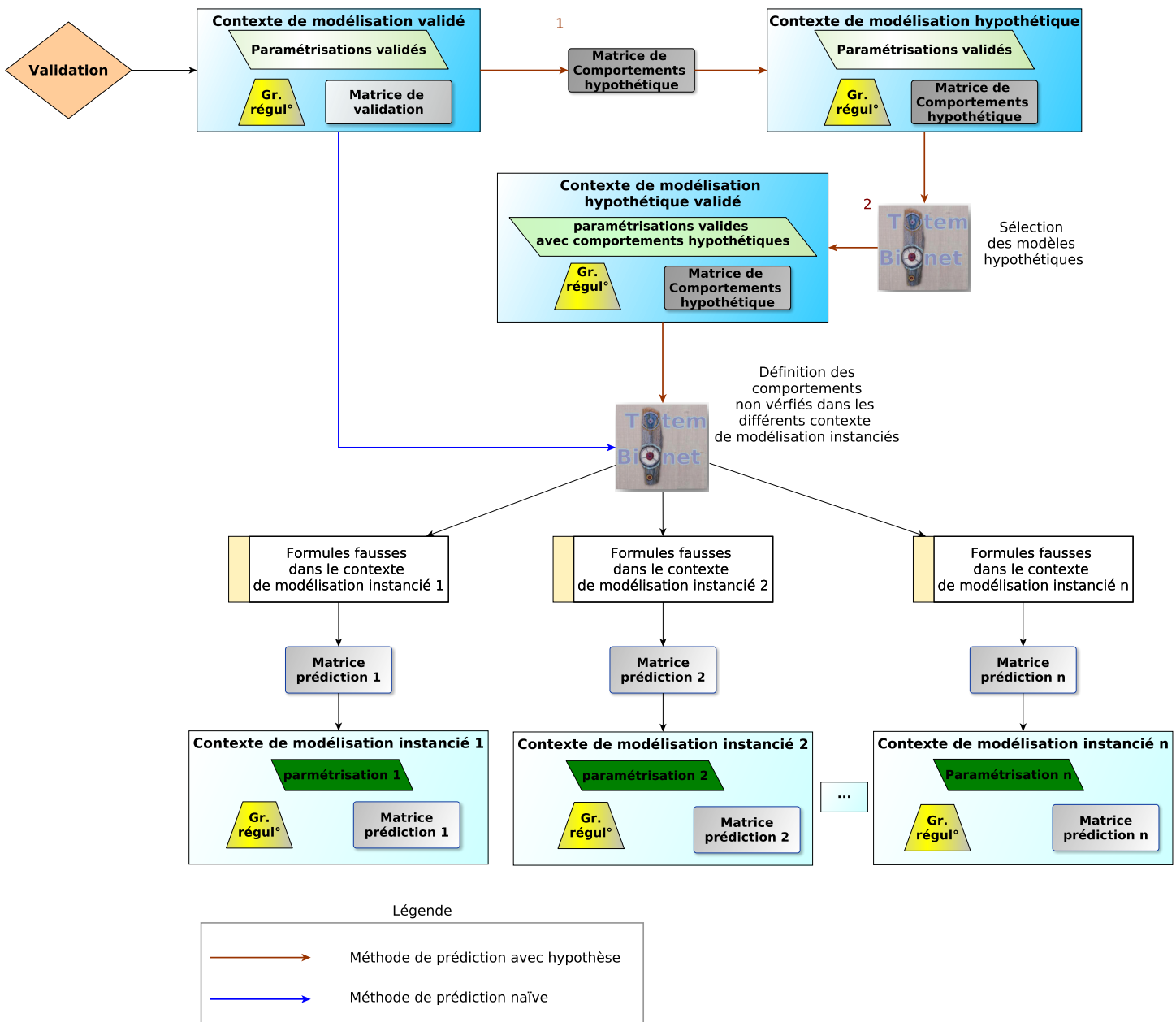


Figure 12.1 – Workflow de la prédiction naïve de comportements et de la prédiction avec hypothèses. Les flèches noires sont des étapes communes aux deux méthodes de prédictions. En bleu sont représentées les étapes de la prédiction naïve et en rouge les étapes de la prédiction par hypothèse.

12.2.3 Prédiction par classification de modèles

Plus récemment, nous avons exploré une troisième méthode qui conserverait les avantages des deux premières, en essayant de jouer sur l'exhaustivité de la méthode naïve tout en limitant le nombre de matrices de prédictions à représenter *via* la classification des modèles. Dans cette

méthode la première étape va prédire tous les comportements pour l'ensemble des contextes de modélisation instanciés. Ensuite ces modèles seront classifiés en fonction des comportements prédits. La première étape utilise *TotemBioNet* pour prédire l'ensemble des comportements pour tous les modèles validés et la seconde utilise LFIT¹ [Ribeiro et al., 2021a].

LFIT (Learning From Interpretation Transition) est un cadre de programmation logique inductive [Apt, 1990] qui représente les systèmes dynamiques multivalués discrets sans mémoire en tant que programmes logiques.

Les premiers travaux proposés dans l'article [Ribeiro et al., 2021b] utilisent cet outil pour aborder notamment le diagnostique de traces d'évènements discrets sur des systèmes biologiques. L'outil est capable d'extraire les caractères minimaux que la cellule doit avoir pour obtenir une trace d'évènements particuliers. Sur ce même concept, nous avons effectué quelques tests où, à partir des paramétrisations des différents modèles valides et de leurs comportements associés, LFIT apprend des règles logiques pour expliquer des paramétrisations d'intérêt. Autrement dit, l'outil permet à partir d'un comportement ou ensemble de comportements sélectionnés de retrouver la totalité des modèles ayant ces propriétés.

L'avantage principal de cette troisième méthode de prédiction réside dans sa capacité à effectuer une étude comparative entre les modèles, vis à vis de la dynamique globale (des comportements) mais aussi vis à vis de la dynamique locale. En effet, pour chaque groupe de modèles exhibant un certain comportement, LFIT est capable de donner une vision synthétique des paramètres importants et de leurs valeurs menant au le comportement donné. En ce sens, l'information est triée et l'analyse de la dynamique locale *via* les valeurs des paramètres donnés par LFIT peut permettre de mettre de côté certains groupes de modèles, lorsqu'il y a incohérence entre les connaissances et les prédictions locales.

L'étape de prédiction et de sélection de modèles complexe devient est une des plus complexes puisqu'elle repose sur la collaboration étroite entre les biologistes et modélisateurs. Elle nécessiterait encore un travail au niveau IHM (Interface Homme Machine) pour mettre en place des représentations compréhensibles et analysables facilement par les biologistes. C'est aussi une des phases de la méthodologie d'enrichissement la plus récente. Les travaux en cours avec LFIT sont prometteurs et permettent d'envisager une automatisation partielle du processus de prédiction. Il reste néanmoins encore plusieurs adaptations à faire pour avoir un pipeline utilisable.

12.2.4 Réduction supplémentaire de la combinatoire pour la validation de modèles

Dans le chapitre 7 nous montrons une méthode de validation basée sur l'utilisation des environnements pour réduire l'explosion combinatoire du nombre de paramétrisations. Bien que cette méthode permette une réduction importante du nombre de paramétrisations, le concept de réduction du graphe de régulation se limite aux variables *directement* impactées par l'environnement. Travailler sur la suppression des arcs asymptotiquement non fonctionnels liant les variables internes des graphes de régulation dans les différents environnements du système biologique représenté serait une solution intéressante pour réduire encore le nombre de paramétrisations et ainsi permettre l'analyse formelle de grands modèles. La suppression des régulations inopérantes dans un environnement pourraient se faire de proche en proche dans le graphe de régulation. La

1. disponible sur le GIT : <https://github.com/Tony-sama/pylfit>

mise en place de cet algorithme nécessite cependant de formaliser les conditions dans lesquelles peuvent s'appliquer ces réductions, en particulier lorsque le graphe de régulation considéré est très inter-connecté avec des multiplexes dont les formules logiques associées font intervenir plusieurs autres variables internes.

Pour conclure, la mise au point de modèles de plus en plus enrichis engendre une augmentation de la taille des modèles, quel que soit le formalisme utilisé. La mise en place d'un modèle devient donc de plus en plus longue. Cette thèse propose une première direction pour contourner au moins partiellement la malédiction de la dimension. En effet, outre le fait de proposer différentes approches dans un cadre applicatif, elle propose une nouvelle méthode de développement de modèles qui pourrait s'appliquer à d'autres cadres de modélisation de systèmes complexes formels moyennant quelques adaptations : cette méthode tend à poser les premières pierres pour que l'activité de modélisation soit de plus en plus accessible dans un cadre d'ingénierie et mène à des modèles de référence collectifs adaptables.

Bibliographie

- [Anderson et al., 2018] Anderson, N. M., Mucka, P., Kern, J. G., and Feng, H. (2018). The emerging role and targetability of the TCA cycle in cancer metabolism. *Protein Cell*, 9(2) :216–237.
- [Andreasson et al., 2021] Andreasson, C., Ansari, D., Ekblom, F., and Andersson, R. (2021). Macropinocytosis : the Achilles’ heel of pancreatic cancer? *Scandinavian Journal of Gastroenterology*, 56(2) :177–179. Publisher : Taylor & Francis _eprint : <https://doi.org/10.1080/00365521.2020.1855471>.
- [Aoyama and Nakaki, 2012] Aoyama, K. and Nakaki, T. (2012). Inhibition of GTRAP3-18 may increase neuroprotective glutathione (GSH) synthesis. 13 :12017–35.
- [Apt, 1990] Apt, K. R. (1990). Logic programming. *Handbook of Theoretical Computer Science, Volume B : Formal Models and Semantics (B)*, 1990 :493–574.
- [Axelrod, 1967] Axelrod, B. (1967). Chapter 3 - Glycolysis. In Greenberg, D. M., editor, *Metabolic Pathways (Third Edition)*, pages 112–145. Academic Press.
- [Baldan et al., 2010] Baldan, P., Cocco, N., Marin, A., and Simeoni, M. (2010). Petri nets for modelling metabolic pathways : a survey. 9(4) :955–989.
- [Batt et al., 2004] Batt, G., Bergamini, D., de Jong, H., Garavel, H., and Mateescu, R. (2004). Model checking genetic regulatory networks using GNA and CADP. In Graf, S. and Mounier, L., editors, *Model Checking Software*, Lecture Notes in Computer Science, pages 158–163. Springer.
- [Berg et al., 2002] Berg, J. M., Tymoczko, J. L., Stryer, L., Berg, J. M., Tymoczko, J. L., and Stryer, L. (2002). *Biochemistry*. W H Freeman, 5th edition.
- [Bernot et al., 2019] Bernot, G., Comet, J.-P., Khalis, Z., Richard, A., and Roux, O. F. (2019). A genetically modified Hoare logic. *Theoretical Computer Science*, 765 :145–157. <https://doi.org/10.1016/j.tcs.2018.02.003>.
- [Bernot et al., 2004] Bernot, G., Comet, J.-P., Richard, A., and Guespin, J. (2004). Application of formal methods to biological regulatory networks : extending thomas’ asynchronous logical approach with temporal logic. 229(3) :339–347.
- [Bi et al., 2021] Bi, Y., Lei, X., Chai, N., and Linghu, E. (2021). NOX4 : a potential therapeutic target for pancreatic cancer and its mechanism. *Journal of Translational Medicine*, 19(1) :515.
- [Boehm, 1988] Boehm, B. W. (1988). A spiral model of software development and enhancement. *Computer*, 21(5) :61–72. Conference Name : Computer.
- [Boghigian et al., 2010] Boghigian, B. A., Seth, G., Kiss, R., and Pfeifer, B. A. (2010). Metabolic flux analysis and pharmaceutical production. *Metabolic Engineering*, 12(2) :81–95.
- [Bouguéon et al., 2021] Bouguéon, M., Boutillier, P., Feret, J., Hazard, O., and Théret, N. (2021). The rule-based model approach. A Kappa model for hepatic stellate cells activation by TGFB1. page 1.
- [Boutillier et al., 2018] Boutillier, P., Maasha, M., Li, X., Medina-Abarca, H. F., Krivine, J., Feret, J., Cristescu, I., Forbes, A. G., and Fontana, W. (2018). The kappa platform for rule-based modeling. 34(13) :i583–i592.

- [Boyenva et al., 2020] Boyenva, D., Bernot, G., Collavizza, H., and Comet, J.-P. (2020). What is a cell cycle checkpoint? the totembionet answer. In *Proceedings of the 18th International Conference on Computational Methods in Systems Biology (CMSB)*, volume 12314 of *LNCS*, pages 362–372, online.
- [Boyer, 2004] Boyer, F. (2004). Reconstruction ab initio de voies métaboliques - Formalisation et approches combinatoires.
- [Broder et al., 2016] Broder, U. N., Jaeger, T., and Jenal, U. (2016). LadS is a calcium-responsive kinase that induces acute-to-chronic virulence switch in *Pseudomonas aeruginosa*. *Nat Microbiol*, 2 :16184.
- [Bryant et al., 2014] Bryant, K. L., Mancias, J. D., Kimmelman, A. C., and Der, C. J. (2014). KRAS : feeding pancreatic cancer proliferation. *Trends in Biochemical Sciences*, 39(2) :91–100.
- [Buscail et al., 2020] Buscail, L., Bournet, B., and Cordelier, P. (2020). Role of oncogenic KRAS in the diagnosis, prognosis and treatment of pancreatic cancer. *Nature Reviews Gastroenterology & Hepatology*, 17(3) :153–168. Number : 3 Publisher : Nature Publishing Group.
- [Cairns et al., 2011] Cairns, R. A., Harris, I. S., and Mak, T. W. (2011). Regulation of cancer cell metabolism. *Nat Rev Cancer*, 11(2) :85–95. Number : 2 Publisher : Nature Publishing Group.
- [Calzone et al., 2006] Calzone, L., Fages, F., and Soliman, S. (2006). BIOCHAM : An environment for modeling biological systems and formalizing experimental knowledge. 22 :1805–7.
- [Cao et al., 2020] Cao, L., Wu, J., Qu, X., Sheng, J., Cui, M., Liu, S., Huang, X., Xiang, Y., Li, B., Zhang, X., and Cui, R. (2020). Glycometabolic rearrangements–aerobic glycolysis in pancreatic cancer : causes, characteristics and clinical applications. *Journal of Experimental & Clinical Cancer Research*, 39.
- [Cimatti et al., 2002] Cimatti, A., Clarke, E., Giunchiglia, E., Giunchiglia, F., Pistore, M., Roveri, M., Sebastiani, R., and Tacchella, A. (2002). NuSMV 2 : An OpenSource Tool for Symbolic Model Checking. In Brinksma, E. and Larsen, K. G., editors, *Computer Aided Verification*, Lecture Notes in Computer Science, pages 359–364. Springer Berlin Heidelberg.
- [Clarke and Emerson, 1982] Clarke, E. M. and Emerson, E. A. (1982). Design and synthesis of synchronization skeletons using branching time temporal logic. In Kozen, D., editor, *Logics of Programs*, Lecture Notes in Computer Science, pages 52–71. Springer.
- [da Veiga Moreira et al.,] da Veiga Moreira, J., Hamraz, M., Abolhassani, M., Schwartz, L., Jollicœur, M., and Peres, S. Metabolic therapies inhibit tumor growth in vivo and in silico. *Sci. Rep.*, 9(1) :3153.
- [Dang, 2012] Dang, C. V. (2012). Links between metabolism and cancer. *Genes Dev.*, 26(9) :877–890.
- [Dardare et al., 2020] Dardare, J., Witz, A., Merlin, J.-L., Gilson, P., and Harlé, A. (2020). SMAD4 and the TGF- β Pathway in Patients with Pancreatic Ductal Adenocarcinoma. *International Journal of Molecular Sciences*, 21(10) :3534.
- [de Alteriis et al., 2018] de Alteriis, E., Carteni, F., Parascandola, P., Serpa, J., and Mazzoleni, S. (2018). Revisiting the Crabtree/Warburg effect in a dynamic perspective : a fitness advantage against sugar-induced cell death. *Cell Cycle*, 17(6) :688–701.
- [DeBerardinis and Thompson, 2012] DeBerardinis, R. J. and Thompson, C. B. (2012). Cellular metabolism and disease : what do metabolic outliers teach us? *Cell*, 148(6) :1132–1144.

- [Fesinmeyer et al., 2005] Fesinmeyer, M. D., Austin, M. A., Li, C. I., De Roos, A. J., and Bowen, D. J. (2005). Differences in Survival by Histologic Type of Pancreatic Cancer. *Cancer Epidemiology, Biomarkers & Prevention*, 14(7) :1766–1773.
- [Folschette, 2019a] Folschette, M. (2019a). The Hoare-foI Tool. Technical Report, Univ. Lille, CNRS, Centrale Lille, UMR 9189 - CRIStAL - Centre de Recherche en Informatique Signal et Automatique de Lille, F-59000 Lille, France.
- [Folschette, 2019b] Folschette, M. (2019b). The Hoare-foI Tool. Technical report, Univ. Lille, CNRS, Centrale Lille, UMR 9189 - CRIStAL - Centre de Recherche en Informatique Signal et Automatique de Lille, F-59000 Lille, France.
- [Forman et al., 2009] Forman, H. J., Zhang, H., and Rinna, A. (2009). Glutathione : Overview of its protective roles, measurement, and biosynthesis. *Mol. Aspects Med.*, 30(1) :1–12.
- [Gibart et al., 2021a] Gibart, L., Bernot, G., Collavizza, H., and Comet, J.-P. (2021a). Totem-bionet enrichment methodology : Application to the qualitative regulatory network of the cell metabolism. In *Proceedings of the 12th International Conference on Bioinformatics Models, Methods and Algorithms*.
- [Gibart et al., 2021b] Gibart, L., Collavizza, H., and Comet, J.-P. (2021b). Greening R. Thomas’ Framework with Environment Variables : A Divide and Conquer Approach. 12881 :36–56. Series Title : Lecture Notes in Computer Science.
- [Gibart et al., 2022] Gibart, L., Collavizza, H., and Comet, J.-P. (2022). A phenotypic matrix for greening qualitative regulatory networks with environments. *submitted*.
- [Gibart et al., 2021c] Gibart, L., Khoodeeram, R., Bernot, G., Comet, J.-P., and Trosset, J.-Y. (2021c). Regulation of eukaryote metabolism : An abstract model explaining the warburg/crabtree effect. *Processes*, 9 :1496.
- [Gombert and Nielsen, 2000] Gombert, A. K. and Nielsen, J. (2000). Mathematical modelling of metabolism. *Current Opinion in Biotechnology*, 11(2) :180–186.
- [Green and MacLennan, 1967] Green, D. E. and MacLennan, D. H. (1967). Chapter 2 - The Mitochondrial System of Enzymes. In Greenberg, D. M., editor, *Metabolic Pathways (Third Edition)*, pages 47–111. Academic Press.
- [Gregg and Prchal, 2018] Gregg, X. T. and Prchal, J. T. (2018). Chapter 44 - Red Blood Cell Enzymopathies. In Hoffman, R., Benz, E. J., Silberstein, L. E., Heslop, H. E., Weitz, J. I., Anastasi, J., Salama, M. E., and Abutalib, S. A., editors, *Hematology (Seventh Edition)*, pages 616–625. Elsevier.
- [Hoare, 1969] Hoare, C. A. R. (1969). An axiomatic basis for computer programming. 12(10) :576–580.
- [Houten and Wanders, 2010] Houten, S. M. and Wanders, R. J. A. (2010). A general introduction to the biochemistry of mitochondrial fatty acid beta-oxidation. *J. Inherit. Metab. Dis.*, 33(5) :469–477.
- [Hsu and Sabatini, 2008] Hsu, P. P. and Sabatini, D. M. (2008). Cancer Cell Metabolism : Warburg and Beyond. *Cell*, 134(5) :703–707.
- [Huth and Ryan, 2000] Huth, M. and Ryan, M. (2000). *Logic in Computer Science : Modelling and reasoning about systems*. Cambridge University Press, Cambridge.
- [Kauffman, 1969] Kauffman, S. A. (1969). Metabolic stability and epigenesis in randomly constructed genetic nets. 22(3) :437–467.

- [Khalis et al., 2009] Khalis, Z., Comet, J.-P., Richard, A., and Bernot, G. (2009). The SMBioNet method for discovering models of gene regulatory networks. *Genes, Genomes and Genomics*, 3(special issue 1) :15–22.
- [Khoodeeram, 2021] Khoodeeram, R. (2021). *Discrete coarse-grained modelling of energy metabolism using formal approach : A study of the dynamics in cell proliferation*. PhD thesis, Université Côte d'Azur.
- [Khoodeeram et al., 2016] Khoodeeram, R., Bernot, G., and Trosset, J.-Y. (2016). *An Ockham Razor model of energy metabolism*, pages 81–101. EDP Sciences, ISBN : 978-2-7598-2116-7.
- [King et al., 2020] King, M. M., Kayastha, B. B., Franklin, M. J., and Patrauchan, M. A. (2020). Calcium Regulation of Bacterial Virulence. *Adv Exp Med Biol*, 1131 :827–855.
- [Kitano, 2002] Kitano, H. (2002). Systems biology : a brief overview. *Science*, 295(5560) :1662–1664.
- [Koch et al., 2017] Koch, I., Náj[then, J., and Schleiff, E. (2017). Modeling the metabolism of arabidopsis thaliana : Application of network decomposition and network reduction in the context of petri nets. 8 :85.
- [Koolman and Röhm, 2012] Koolman, J. and Röhm, K.-H. (2012). *Color Atlas of Biochemistry*. Thieme Publishing Group, Stuttgart, 3rd edition, revised and updated edition.
- [Kumagai, 2013] Kumagai, H. (2013). Amino Acid Production. In Rosenberg, E., DeLong, E. F., Lory, S., Stackebrandt, E., and Thompson, F., editors, *The Prokaryotes : Applied Bacteriology and Biotechnology*, pages 169–177. Springer, Berlin, Heidelberg.
- [Liberti and Locasale, 2016] Liberti, M. V. and Locasale, J. W. (2016). The Warburg Effect : How Does it Benefit Cancer Cells? *Trends in biochemical sciences*, 41(3) :211–218.
- [Lowenstein, 1967] Lowenstein, J. M. (1967). Chapter 4 - The Tricarboxylic Acid Cycle. In Greenberg, D. M., editor, *Metabolic Pathways (Third Edition)*, pages 146–270. Academic Press.
- [Malhotra et al., 2019] Malhotra, S., Hayes, D., and Wozniak, D. J. (2019). Cystic Fibrosis and Pseudomonas aeruginosa : the Host-Microbe Interface. *Clinical Microbiology Reviews*, 32(3).
- [Mansour, 1963] Mansour, T. E. (1963). Studies on Heart Phosphofructokinase : Purification, Inhibition, and Activation. *J. Biol. Chem.*, 238(7) :2285–2292.
- [Martin et al., 1993] Martin, D. W., Schurr, M. J., Mudd, M. H., and Deretic, V. (1993). Differentiation of Pseudomonas aeruginosa into the alginate-producing form : inactivation of mucB causes conversion to mucoidy. *Mol Microbiol*, 9(3) :497–506.
- [Martinez-Useros et al., 2017] Martinez-Useros, J., Li, W., Cabeza-Morales, M., and Garcia-Foncillas, J. (2017). Oxidative Stress : A New Target for Pancreatic Cancer Prognosis and Treatment. *Journal of Clinical Medicine*, 6(3) :29.
- [Maus and Peters, 2017] Maus, A. and Peters, G. J. (2017). Glutamate and alpha-ketoglutarate : key players in glioma metabolism. *Amino Acids*, 49(1) :21–32.
- [McKeehan, 1986] McKeehan, W. L. (1986). Glutaminolysis in animal cells. In Morgan, M. J., editor, *Carbohydrate Metabolism in Cultured Cells*, pages 111–150. Springer US.
- [Molenaar et al., 2009] Molenaar, D., van Berlo, R., de Ridder, D., and Teusink, B. (2009). Shifts in growth strategies reflect tradeoffs in cellular economics. *Molecular Systems Biology*, 5(1) :323. Publisher : John Wiley & Sons, Ltd.

- [Moreira et al., 2016] Moreira, J. d. V., Hamraz, M., Abolhassani, M., Bigan, E., Pérès, S., Paulevé, L., Nogueira, M. L., Steyaert, J.-M., and Schwartz, L. (2016). The redox status of cancer cells supports mechanisms behind the warburg effect. *Metabolites*, 6(4).
- [Naldi et al., 2018] Naldi, A., Hernandez, C., Levy, N., Stoll, G., Monteiro, P. T., Chaouiya, C., Helikar, T., Zinovyev, A., Calzone, L., Cohen-Boulakia, S., Thieffry, D., and Paulevé, L. (2018). The CoLoMoTo Interactive Notebook : Accessible and Reproducible Computational Analyses for Qualitative Biological Networks. *Front. Physiol.*, 9.
- [Naldi et al., 2007] Naldi, A., Thieffry, D., and Chaouiya, C. (2007). Decision diagrams for the representation and analysis of logical models of genetic networks. In Calder, M. and Gilmore, S., editors, *Computational Methods in Systems Biology*, pages 233–247, Berlin, Heidelberg. Springer.
- [Nelson and Cox, 2012] Nelson, D. L. and Cox, M. M. (2012). *Lehninger Principles of Biochemistry*. W.H. Freeman and Company, New York, 6 édition edition.
- [Orth et al., 2010] Orth, J. D., Thiele, I., and Palsson, B. A. (2010). What is flux balance analysis? 28(3) :245–248. Bandiera_abtest : a Cg_type : Nature Research Journals Number : 3 Primary_atype : Reviews Publisher : Nature Publishing Group Subject_term : Biochemical reaction networks;Biotechnology;Metabolism Subject_term_id : biochemical-reaction-networks;biotechnology;metabolism.
- [Pandol, 2011] Pandol, S. J. (2011). The Exocrine Pancreas. *Colloquium Series on Integrated Systems Physiology : From Molecule to Function*, 3(1) :1–64. Publisher : Morgan & Claypool Life Sciences.
- [Parkhitko et al., 2019] Parkhitko, A. A., Jouandin, P., Mohr, S. E., and Perrimon, N. (2019). Methionine metabolism and methyltransferases in the regulation of aging and lifespan extension across species. *Aging Cell*, 18(6) :e13034.
- [Petri, 1980] Petri, C. A. (1980). Introduction to general net theory. In Brauer, W., editor, *Net Theory and Applications*, Lecture Notes in Computer Science, pages 1–19. Springer.
- [Pfeiffer and Morley, 2014] Pfeiffer, T. and Morley, A. (2014). An evolutionary perspective on the Crabtree effect. *Front. Mol. Biosci.*, 1.
- [Pommier and Kohn, 2003] Pommier, Y. and Kohn, K. W. (2003). Cycle cellulaire et points de contrôle en oncologie : nouvelles cibles thérapeutiques. *médecine/sciences*, 19(2) :173–186.
- [Printz, 2006] Printz, J. (2006). *Architecture logicielle : concevoir des applications simples, s ? ?res et adaptables*. Dunod, Paris. OCLC : 184969405.
- [Pritt et al., 2007] Pritt, B., O’Brien, L., and Winn, W. (2007). Mucoïd Pseudomonas in Cystic Fibrosis. *American Journal of Clinical Pathology*, 128(1) :32–34.
- [Purohit et al., 2019] Purohit, V., Simeone, D. M., and Lyssiotis, C. A. (2019). Metabolic Regulation of Redox Balance in Cancer. *Cancers (Basel)*, 11(7).
- [Raman and Chandra, 2009] Raman, K. and Chandra, N. (2009). Flux balance analysis of biological systems : applications and challenges. *Briefings in Bioinformatics*, 10(4) :435–449.
- [Reddy et al., 1993] Reddy, V. N., Mavrovouniotis, M. L., and Liebman, M. N. (1993). Petri net representations in metabolic pathways. *Proceedings. International Conference on Intelligent Systems for Molecular Biology*, 1 :328–336.

- [Reiser et al., 2001] Reiser, P. G. K., King, R. D., Kell, D. B., Muggleton, S. H., Bryant, C. H., and Oliver, S. G. (2001). Developing a logical model of yeast metabolism. *Electronic Transactions in Artificial Intelligence*, 5 :233–244.
- [Reitzer, 2009] Reitzer, L. (2009). Amino Acid Synthesis. In Schaechter, M., editor, *Encyclopedia of Microbiology (Third Edition)*, pages 1–17. Academic Press, Oxford.
- [Resat et al., 2009] Resat, H., Petzold, L., and Pettigrew, M. F. (2009). Kinetic modeling of biological systems. *Methods in Molecular Biology (Clifton, N.J.)*, 541 :311–335.
- [Ribeiro et al., 2021a] Ribeiro, T., Folschette, M., Magnin, M., and Inoue, K. (2021a). Learning any memory-less discrete semantics for dynamical systems represented by logic programs. *Mach Learn.*
- [Ribeiro et al., 2021b] Ribeiro, T., Folschette, M., Magnin, M., and Inoue, K. (2021b). Polynomial Algorithm For Learning From Interpretation Transition. In *1st International Joint Conference on Learning & Reasoning*, (virtual), Greece.
- [Richard, 2008] Richard, A. (2008). Fair paths in CTL. Technical report, available at <https://gitlab.com/totembionet/totembionet>.
- [Rigoulet et al., 2020] Rigoulet, M., Bouchez, C. L., Paumard, P., Ransac, S., Cuvellier, S., Duvezin-Caubet, S., Mazat, J. P., and Devin, A. (2020). Cell energy metabolism : An update. *Biochim. Biophys. Acta Bioenerg.*, 1861(11) :148276.
- [Rui et al., 2015] Rui, B., Yi, Y., Shen, T., Zheng, M., Zhou, W., Du, H., Fan, Y., Wang, Y., Zhang, Z., Xu, S., Liu, Z., Wen, H., and Xie, X. (2015). Elementary flux mode analysis revealed cyclization pathway as a powerful way for NADPH regeneration of central carbon metabolism. 10(6) :e0129837. Publisher : Public Library of Science.
- [Salnikov et al., 2012] Salnikov, A. V., Liu, L., Platen, M., Gladkikh, J., Salnikova, O., Ryschich, E., Mattern, J., Moldenhauer, G., Werner, J., Schemmer, P., Büchler, M. W., and Herr, I. (2012). Hypoxia induces EMT in low and highly aggressive pancreatic tumor cells but only cells with cancer stem cell characteristics acquire pronounced migratory potential. *PloS One*, 7(9) :e46391.
- [Sarkisova et al., 2005] Sarkisova, S., Patrauchan, M. A., Berglund, D., Nivens, D. E., and Franklin, M. J. (2005). Calcium-induced virulence factors associated with the extracellular matrix of mucoid *Pseudomonas aeruginosa* biofilms. *J Bacteriol*, 187(13) :4327–4337.
- [Schuster et al., 2000] Schuster, S., Fell, D. A., and Dandekar, T. (2000). A general definition of metabolic pathways useful for systematic organization and analysis of complex metabolic networks. *Nature Biotechnology*, 18(3) :326–332. Number : 3 Publisher : Nature Publishing Group.
- [Schuster and Hilgetag,] Schuster, S. and Hilgetag, C. On elementary flux modes in biochemical reaction systems at steady state. 02(2) :165–182. Publisher : World Scientific Publishing Co.
- [Shi and Tu, 2015] Shi, L. and Tu, B. P. (2015). Acetyl-CoA and the Regulation of Metabolism : Mechanisms and Consequences. *Curr. Opin. Cell Biol.*, 33 :125–131.
- [Snoussi and Thomas, 1993] Snoussi, E. and Thomas, R. (1993). Logical identification of all steady states : The concept of feedback loop characteristic states. *Bulletin of Mathematical Biology*, 55(5) :973–991.
- [Sober, 2015] Sober, E. (2015). *Ockham’s Razors : A User’s Manual*. Cambridge University Press.

- [Son et al., 2013] Son, J., Lyssiotis, C. A., Ying, H., Wang, X., Hua, S., Ligorio, M., Perera, R. M., Ferrone, C. R., Mullarky, E., Shyh-Chang, N., Kang, Y., Fleming, J. B., Bardeesy, N., Asara, J. M., Haigis, M. C., DePinho, R. A., Cantley, L. C., and Kimmelman, A. C. (2013). Glutamine supports pancreatic cancer growth through a Kras-regulated metabolic pathway. *Nature*, 496(7443) :101–105.
- [Stein and Stein, 1967] Stein, O. and Stein, Y. (1967). Lipid synthesis, intracellular transport, storage, and secretion. *J. Cell Biol.*, 33(2) :319–339.
- [Storz, 2006] Storz, P. (2006). KRas, ROS and the initiation of pancreatic cancer. 8(1) :38–42.
- [Sullivan et al., 2021] Sullivan, M. R., Darnell, A. M., Reilly, M. F., Kunchok, T., Joesch-Cohen, L., Rosenberg, D., Ali, A., Rees, M. G., Roth, J. A., Lewis, C. A., and Vander Heiden, M. G. (2021). Methionine synthase is essential for cancer cell proliferation in physiological folate environments. *Nature Metabolism*, 3(11) :1500–1511. Number : 11 Publisher : Nature Publishing Group.
- [Terry et al., 1991] Terry, J. M., Piña, S. E., and Mattingly, S. J. (1991). Environmental conditions which influence mucoid conversion Pseudomonas aeruginosa PAO1. *Infection and Immunity*, 59(2) :471–477. Publisher : American Society for Microbiology Journals Section : Research Article.
- [Thomas, 1973a] Thomas, R. (1973a). Boolean formalization of genetic control circuits. *J. Theor. Biol.*, 42(3) :563–585.
- [Thomas, 1973b] Thomas, R. (1973b). Boolean formalization of genetic control circuits. *J. Theor. Biol.*, 42(3) :563–585.
- [Thomas, 1978] Thomas, R. (1978). Logical analysis of systems comprising feedback loops. *J. Theor. Biol.*, 73(4) :631–56.
- [Vaitheesvaran et al., 2015] Vaitheesvaran, B., Xu, J., Yee, J., Lu, Q.-Y., Go, V. L., Xiao, G. G., and Lee, W.-N. (2015). The Warburg effect : a balance of flux analysis. *Metabolomics*, 11(4) :787–796.
- [Vaquero et al., 2004] Vaquero, E. C., Edderkaoui, M., Pandol, S. J., Gukovsky, I., and Gukovskaya, A. S. (2004). Reactive oxygen species produced by NAD(P)H oxidase inhibit apoptosis in pancreatic cancer cells. *The Journal of Biological Chemistry*, 279(33) :34643–34654.
- [Varma and Palsson,] Varma, A. and Palsson, B. O. Metabolic flux balancing : Basic concepts, scientific and practical use. 12(10) :994–998. Number : 10 Publisher : Nature Publishing Group.
- [Wang, 2013] Wang, R.-S. (2013). Ordinary differential equation (ODE), model. In Dubitzky, W., Wolkenhauer, O., Cho, K.-H., and Yokota, H., editors, *Encyclopedia of Systems Biology*, pages 1606–1608. Springer.
- [Waters and Der, 2018] Waters, A. M. and Der, C. J. (2018). KRAS : The Critical Driver and Therapeutic Target for Pancreatic Cancer. *Cold Spring Harbor Perspectives in Medicine*, 8(9) :a031435.
- [Wilson, 2017] Wilson, D. F. (2017). Oxidative phosphorylation : regulation and role in cellular and tissue metabolism. *The Journal of Physiology*, 595(23) :7023–7038.
- [Wu et al., 2016] Wu, N., Yang, M., Gaur, U., Xu, H., Yao, Y., and Li, D. (2016). Alpha-Ketoglutarate : Physiological Functions and Applications. *Biomol. Ther. (Seoul)*, 24(1) :1–8.

- [Yaku et al., 2018] Yaku, K., Okabe, K., Hikosaka, K., and Nakagawa, T. (2018). NAD Metabolism in Cancer Therapeutics. *Frontiers in Oncology*, 8.
- [Yang et al., 2017] Yang, L., Venneti, S., and Nagrath, D. (2017). Glutaminolysis : A hallmark of cancer metabolism. *Annu. Rev. Biomed. Eng.*, 19 :163–194.
- [Yin et al., 2012] Yin, C., Qie, S., and Sang, N. (2012). Carbon source metabolism and its regulation in cancer cells. *Crit. Rev. Eukaryot. Gene Expr.*, 22(1) :17–35.
- [Zeitouni et al., 2016] Zeitouni, D., Pylayeva-Gupta, Y., Der, C. J., and Bryant, K. L. (2016). KRAS Mutant Pancreatic Cancer : No Lone Path to an Effective Treatment. *Cancers (Basel)*, 8(4).
- [Zhu and Thompson, 2019] Zhu, J. and Thompson, C. B. (2019). Metabolic regulation of cell growth and proliferation. *Nat. Rev. Mol. Cell. Biol.*, 20(7) :436–450.

Pages web

- [1] Cancer du pancréas - symptômes, causes, traitements et prévention. Section : Santé.
- [2] Pancreatic Adenocarcinoma Therapeutic Targets Revealed by Tumor-Stroma Cross-Talk Analyses in Patient-Derived Xenografts. l.
- [3] V-Model - an overview | ScienceDirect Topics, 2008.
- [4] Pancreatic cancer | CRCM, December 2018.
- [5] Présentation du CRCM | CRCM, March 2019.
- [6] The International Agency for Research on Cancer (IARC). Global Cancer Observatory.
- [7] Fondation ARC pour la recherche sur le cancer. Les cancers du pancréas | Fondation ARC pour la recherche sur le cancer.

Liste des figures

2.1	Exemple d'un réseau métabolique. En rouge sont les métabolites d'entrée, en orange ceux de sortie. En vert sont les enzymes des réactions. R1, R2, R3, R4 nomment les réactions du réseau métabolique.	8
2.2	Exemple d'un graphe de régulation. Les entités X et Y représentent des éléments abstraits de voies métaboliques.	8
2.3	Figure représentant les différents éléments d'un réseau de Petri.	16
2.4	1) Réseau booléen sur trois variables X , Y et Z . 2) Fonctions booléennes exprimant différents types de coopération. 3) Tables de vérité pour chacun des nœuds. . . .	19
2.5	Graphe de transitions des états pour le réseau booléen donné figure 2.4. En rouge la transition de l'état $(0,0,1)$ à l'état $(0,0,0)$	20
3.1	Graphe de régulation pour le système de production de mucus chez <i>Pseudomonas aeruginosa</i> . Les variables sont représentées par des cercles et les multiplexes par des rectangles.	24
3.2	Deux configurations possibles de prédécesseurs pour <i>MucB</i>	24
3.3	Graphe des états pour <i>Pseudomonas aeruginosa</i>	26
3.4	Schéma de la variable opéron et de l'ensemble de ses prédécesseurs	26
3.5	graphe de transitions pour <i>Pseudomonas aeruginosa</i> d'après la table 3.3.	30
3.6	Deux exemples de cycles positifs.	31
3.7	Graphes de transitions possibles associés aux graphes de régulation de la figure 3.6 (cycle positifs). En vert (resp. noir) le graphe de transitions associé au graphe de régulation vert (resp. noir).	32
3.8	Deux exemple de cycles négatifs.	32
3.9	Graphes de transitions associés aux graphes de régulation de la figure 3.8 (cycle négatifs).	32
3.10	Schéma récapitulatif des différents connecteurs temporels de <i>CTL</i> . Issu du cours en ligne de Antonio Gonzàles Bergueño pour l'université d'Oslo, https://www.uio.no/studier/emner/matnat/ifi/INF5140/v17/slides/clt-antonio.pdf	34
3.11	Graphe de transitions des états pour <i>Pseudomonas aeruginosa</i> . Les cases rouges (<i>Operon</i> =2) indiquent les états à ne jamais atteindre pour satisfaire la formule, tout en partant des cases vertes (<i>Operon</i> =0).	36
3.12	Graphe de transitions montrant la non équitabilité des chemins en <i>CTL</i> . Le chemin infini qui boucle sur les états blancs n'atteindra jamais un état où $x=2$	36
4.1	Cycle de vie en « V » adapté au développement de modèles de systèmes dynamiques. En bleu les objets formels construits, en noir les activités et en jaune les analogies avec le génie logiciel.	41
4.2	Schéma d'un contexte de modélisation	42
4.3	Cycle de vie en spirale pour le développement de modèles enrichis	44

4.4	Méthodologie incrémentale d'enrichissement de modèles.	46
6.1	Visualisation des graphes de régulation locaux pour les paramètres K_{MucB} et $K_{MucB,prod}$ respectivement.	60
6.2	Workflow de <i>TotemBioNet</i> traduit et issu de l'article [Boyenval et al., 2020].	64
6.3	Fichier d'entrée de <i>TotemBioNet</i> pour le modèle de <i>Pseudomonas aeruginosa</i> avec logique de Hoare, CTL et <i>Fair-Path CTL</i>	65
7.1	Méthodologie globale. Les autorégulations sur les variables d'environnement permettent alors de fixer les variables d'environnements, et ainsi de simuler les comportements différents pour chacun des environnements.	67
7.2	Schéma des étapes de l'approche diviser pour régner. Chaque environnement est étudié séparément, menant à un ensemble de modèles cohérent spécifiques. La dernière étape consiste à construire les modèles globaux à partir des modèles cohérents aux différents environnements.	68
7.3	réseau de régulation du modèle de production de mucus par <i>Pseudomonas aeruginosa</i> avec autorégulation sur la variable <i>Calcium</i> (indiquée en vert). Les paramètres sont indiqués en syntaxe concrète de <i>TotemBioNet</i>	70
7.4	Graphe de transitions pour <i>Pseudomonas aeruginosa</i> avec l'autorégulation pour simuler les environnements de <i>Calcium</i> . En vert les états où <i>Calcium=0</i>	71
7.5	GTA pour <i>Pseudomonas aeruginosa</i> dans l'environnement e^0	75
7.6	<i>GreenBioNet</i> Workflow : automatisation de l'approche « diviser pour régner » pour obtenir le modèle d'une matrice de validation.	81
7.7	Fichiers d'entrée donnés à <i>TotemBioNet</i> pour la validation du modèle de <i>Pseudomonas aeruginosa</i>	84
8.1	Réseau de régulation de <i>Pseudomonas aeruginosa</i> présenté chapitre 3. Les paramètres sont indiqués dans les rectangles en pointillés en syntaxe concrète de <i>TotemBioNet</i>	87
8.2	Graphe de régulation de <i>Pseudomonas aeruginosa</i> avec ajout d'un multiplexe régulateur positif hypo	88
8.3	Graphe de régulation après ajout d'un multiplexe inhibiteur hyponeg	89
8.4	Graphe de régulation de production de mucus <i>Pseudomonas aeruginosa</i> avant enrichissement par le Calcium	92
8.5	Graphe de régulation du modèle booléen de la production de mucus chez <i>Pseudomonas aeruginosa</i>	95
8.6	Représentation des évolutions possibles des précédents niveaux après les différents enrichissements possibles. En jaune le niveau 0 avant enrichissement, en bleu le niveau 1 avant enrichissement. Après enrichissement le sens du niveau non dédoublé reste identique mais la valeur associé peut être amenée à changer.	95
8.7	Graphe de régulation en cours de construction avec les variables filles et leurs régulations en aval	100
8.8	Graphe de régulation en cours de construction avec les variables filles et leurs régulations en aval	102
9.1	Carte simplifiée des voies métaboliques issue de l'article [Boyer, 2004].	110
9.2	Schéma des réactions du cycle de Krebs issu de l'article [Maus and Peters, 2017]	111

9.3	Graphe de régulation du métabolisme de la cellule. Les cercles représentent des variables alors que les rectangles représentent des multiplexes. Les entités bleues font référence à la biomasse, les jaunes aux métabolites et les rouges aux voies métaboliques. Les lignes pleines indiquent les cibles de régulation. Les lignes pointillées désignent les sources de régulation (mathématiquement inutiles puisqu'elles peuvent être déduites de formules des multiplexes). Dans les formules logiques « ! » signifie la négation, « & » la conjonction et « » la disjonction. . . .	114
9.4	Ressources et cibles de la variable représentant la glycolyse	129
9.5	les ressources et les cibles la variable <i>KREBS</i>	130
9.6	Ressources et cibles de <i>FERM</i>	131
9.7	Ressources et cibles de <i>PHOX</i>	131
9.8	Ressources et cibles de <i>nLBP</i>	131
9.9	Ressources et cibles de <i>LBP</i>	132
9.10	Ressources et cibles de <i>NCD</i>	132
9.11	Ressources et cibles de <i>O2</i>	133
9.12	Ressources et cibles de <i>NADH</i>	135
9.13	Ressources et cibles de <i>ATP</i>	136
10.1	Schéma du fonctionnement de l'interrupteur moléculaire d'une protéine KRAS. Figure adaptée de l'article [Waters and Der, 2018].	142
10.2	Graphe de régulation enrichi représentant la régulation du métabolisme de la cellule mutée ou non pour le gène KRAS. Les entités bleues font référence à la biomasse, les jaunes aux métabolites et les rouges aux voies métaboliques. L'entité orange fait référence à un composé génétique.	144
10.3	Graphe de régulation enrichi avec le multiplexe GLN metabolism et le changement des seuils pour les multiplexes impliquant la variable <i>NADH</i> (encadré vert). Les entités bleues font référence à la biomasse, les jaunes aux métabolites et les rouges aux voies métaboliques. L'entité orange fait référence à un composé génétique.	150
10.4	Réactions de détoxification des ROS par l'intermédiaire du Couple GSH/GSSH. Figure provenant de l'article [Aoyama and Nakaki, 2012].	154
10.5	Graphe de régulation après insertion de la variable <i>ROS</i> et des multiplexes régulateurs NOXc et GSHc (encadré vert).	156
11.1	Voies métaboliques liées à la méthionine. En vert le cycle du Folate, en rose la voie de la transsulfuration, en bleu la voie de sauvetage de la méthionine et en violet la voie de la méthionine <i>de Novo</i> . Les réactions permettant la réduction des ROS sont entourées en rouge. Support construit par Sophie vasseur.	161
11.2	Graphe de régulation enrichi avec la variable MET de novo (encadré vert) pour représenter la régulation du métabolisme dans le cadre d'une cellule cancéreuse agressive.	163
11.3	Graphe de régulation après enrichissement avec <i>TGF-β</i>	167
12.1	Workflow de la prédiction naïve de comportements et de la prédiction avec hypothèses. Les flèches noires sont des étapes communes aux deux méthodes de prédictions. En bleu sont représentées les étapes de la prédiction naïve et en rouge les étapes de la prédiction par hypothèse.	174

Liste des tableaux

2.1	Matrice d'incidence calculée à partir du réseau de Petri décrit figure 2.3.	17
3.1	Tableau représentant le réseau de régulation de la production de mucus chez <i>Pseudomonas aeruginosa</i>	25
3.2	Tableau récapitulatif des paramètres applicables pour <i>Pseudomonas aeruginosa</i> . Lorsque MucB est absent alors Operon a free dans ses ressources car c'est un inhibiteur. L'absence d'un inhibiteur joue un rôle similaire à la présence d'un activateur et inversement. La ligne soulignée en vert correspond à l'état (2,0,1) utilisé comme exemple.	27
3.3	Exemple d'une paramétrisation pour <i>Pseudomonas aeruginosa</i> . Le tableau contient l'ensemble des paramètres et points focaux associés. En vert sont représentés les paramètres qui s'appliquent dans l'état (2,0,1) utilisé comme exemple.	29
5.1	Matrice de validation pour le contexte de modélisation associé à la production de mucus par <i>Pseudomonas aeruginosa</i>	56
8.1	Matrice de validation pour le modèle de <i>Pseudomonas aeruginosa</i>	87
8.2	Matrice de validation avant (gauche) et après (droite) enrichissement d'un multiplexe sur la variable d'environnement O . En vert sont les cases qui nécessitent une attention particulière. Les significations des motifs « osc » et « osc(0-1) » sont disponibles chapitre 5 section 5.3.2.	91
8.3	Table des paramètres d' operon après ajout de la variable Calcium . En vert sont les paramètres ajoutés par la présence de calcium comme ressource potentielle.	93
8.4	Matrice de validation pour le modèle de <i>Pseudomonas aeruginosa</i> avant enrichissement avec le Calcium	94
8.5	Matrice de validation pour le modèle de <i>Pseudomonas aeruginosa</i> après enrichissement avec le Calcium . Le fond vert indique les enrichissements faisant partie de la nouvelle matrice enrichie. En blanc les comportements reportés de la matrice d'origine.	94
8.6	Table des paramètres d' operon pour le modèle <i>Pseudomonas aeruginosa</i> booléen. En vert les paramètres à réévaluer.	97
8.7	Table des paramètres d' operon pour le modèle <i>Pseudomonas aeruginosa</i> multivalué.	97
8.8	Exemple fictif de matrice de validation pour le modèle de <i>Pseudomonas aeruginosa</i> booléen. En vert le comportement qui devra être adapté pour le passage du modèle booléen au multivalué.	97
8.9	Matrice de validation pour le modèle de <i>Pseudomonas aeruginosa</i> multivalué. En vert clair le comportement modifié adapté à l'ajout d'un de niveau de la variable operon	98

8.10	Table représentant les régulations <i>en sortie</i> de la variable mère et l'association de chacune d'entre elles avec les variables filles. « \rightarrow <i>prod</i> » de la colonne operon mère indique la régulation de opéron dans le graphe de régulation initial qui régule <i>via</i> le multiplexe prod la variable MucB . La partie gauche du tableau indique la régulation dans le graphe initial et la partie de droite du tableau (après la double barre verticale) indique les variables filles et leurs régulations en aval associées.	100
8.11	Régulations <i>en entrée</i> de la variable mère et des variables filles. A gauche la variable présente avant enrichissement et à droite les variables remplaçant la variable mère.	101
8.12	Régulations <i>en entrée</i> de la variable mère et des variables filles. A gauche la variable présente avant enrichissement et à droite les variables remplaçant la variable mère.	103
8.13	Matrice de validation pour le modèle de <i>Pseudomonas aeruginosa</i> après scission de la variable Operon	105
9.1	Matrice de validation pour le modèle de la régulation « générale » du métabolisme cellulaire. Chaque ligne (resp. colonne) représente une condition environnementale (resp. une variable interne). De plus chaque case de la table formalise le comportement connu d'une variable dans un environnement. Les cases grises indiquent l'absence d'une connaissance ou que la connaissance ne fait pas consensus pour l'ensemble des biologistes. Les lignes 16, 17, 18, 34, 35 et 36 exhibent la glycolyse en aérobie.	123
9.2	Valeurs des paramètres de la section 9.4. Seulement 7 valeurs des paramètres restent libres : $K_{ATP, LBP\ PHOX}$, $K_{ATP, GLYC1\ LBP}$, $K_{ATP, GLYC1\ GLYC2\ LBP}$, $K_{ATP, GLYC1\ LBP\ PHOX}$, $K_{ATP, GLYC1\ GLYC2\ LBP\ PHOX}$, $K_{NADH, FERM\ GLYC\ PHOX}$ et $K_{NADH, FERM\ GLYC\ Krebs}$	128
10.1	Matrice de validation pour le modèle de la régulation du métabolisme cellulaire avec ou sans mutation du gène KRAS. Les lignes $\rightsquigarrow 1$ à $\rightsquigarrow 36$ correspondent aux lignes déjà exhibées dans le modèle général de la régulation du métabolisme. Les lignes $\rightsquigarrow 37$ et $\rightsquigarrow 38$ sont de nouvelles spécifications. Nous avons des cases grises dans l'environnement pour stipuler que la valeur des variables d'environnement pour ces cases n'est pas importante.	146
10.2	Matrice de prédiction pour les lignes $\rightsquigarrow 37$ et $\rightsquigarrow 38$	147
10.3	Paramètres ré-estimés à partir du sens des paramètres du modèle initial après doublement du niveau de NADH.	151
10.4	Nouveaux paramètres de la variable NADH dûs à l'ajout du multiplexe GLN metabolism	151
10.5	Matrice de validation pour le modèle de la régulation du métabolisme avec ajout du multiplexe GLN metabolism et l'ajout d'un niveau à la variable NADH . Les lignes $\rightsquigarrow 1$ à $\rightsquigarrow 36$ correspondent aux lignes déjà exhibées dans le modèle général de la régulation du métabolisme. Les lignes $\rightsquigarrow 37$ et $\rightsquigarrow 38$ sont de nouvelles spécifications.	152

10.6	Matrice de validation pour le modèle de la régulation du métabolisme avec insertion de la variable <i>ROS</i> . Les lignes $\rightsquigarrow 1$ à $\rightsquigarrow 38$ correspondent aux lignes déjà exhibées dans le modèle général de la régulation du métabolisme après l'enrichissement précédent excepté pour la colonne ROS qui constitue de nouvelles spécifications.	157
11.1	Matrice de validation pour le modèle de la régulation du métabolisme avec insertion de la variable <i>Met de novo</i> . Les lignes $\rightsquigarrow 1$ à $\rightsquigarrow 38$ correspondent aux lignes déjà spécifiées dans le modèle général de la régulation du métabolisme après l'enrichissement précédent excepté pour la colonne ROS qui requiert de nouvelles spécifications.	165
11.2	Matrice de validation pour le modèle de la régulation du métabolisme avec insertion de la variable d'environnement <i>TGF-β</i> . Les lignes $\rightsquigarrow 1$ à $\rightsquigarrow 38$ correspondent aux lignes déjà spécifiées dans le modèle général de la régulation du métabolisme après l'enrichissement précédent, excepté pour les colonnes <i>MET de novo</i> et <i>ROS</i> qui constituent de nouvelles spécifications.	168
A.1	Valeurs des paramètres du modèle de la régulation du métabolisme pour une cellule cancéreuse dans le pancréas agressif ou non. 10 valeurs des paramètres restent libres : $K_{ATP, LBP\ PHOX}$, $K_{ATP, GLYC1\ LBP}$, $K_{ATP, GLYC1\ GLYC2\ LBP}$, $K_{ATP, GLYC1\ LBP\ PHOX}$, $K_{ATP, GLYC1\ GLYC2\ LBP\ PHOX}$, $K_{NADH, FERMGLYC\ PHOX}$, $K_{NADH, GLNmFERMGLYC\ PHOX}$, $K_{NADH, FERMGLYC\ Krebs}$, $K_{NADH, GLNmFERMGLYC\ Krebs}$, $K_{ROS, GSHc}$, $K_{ROS, GSHc\ NOX}$, $K_{ROS, GSHa}$, $K_{ROS, GSHa\ GSHc}$	198

Liste des définitions

2.3.1 Réseau de Petri	16
3.1.1 Graphe de régulation avec multiplexes	23
3.2.1 les états	25
3.2.2 Ressources	26
3.2.3 Conditions de Snoussi	27
3.2.4 Paramètre applicable dans un état	27
3.2.5 Paramètres opérables	28
3.2.6 Paramétrisation	28
3.2.7 Point focal d'un état	28
3.2.8 <i>GTA</i> pour une paramétrisation P	30
3.3.1 Langage <i>CTL</i>	33
3.3.2 Modèles de $\varphi \in \mathcal{L}_V$	35
5.1.1 Graphe de régulation avec variables d'environnement	49
5.2.1 Environnements	51
5.2.2 Propriété environnementale	51
5.3.1 Comportement stationnaire	53
5.3.2 Comportement oscillatoire borné	54
5.3.3 Comportement oscillatoire	54
5.4.1 Matrice de phénotypes	55
5.4.2 Propriété environnementale d'une matrice phénotypique	55
7.1.1 Propriété biologique dans un environnement	71
7.2.1 Espace d'états pour un environnement	73
7.2.2 Paramètres opérables pour un environnement	74
7.2.3 Réseau de régulation dans un environnement	74
7.2.4 <i>GTA</i> pour une paramétrisation P_e dans l'environnement e	74
7.2.5 Modèles d'une propriété environnementale	75
7.2.6 Abstraction d'une paramétrisation	76
7.2.7 Abstraction de modèles	76
7.2.8 Modèles de propriétés environnementales	76
7.3.1 Modèle d'une matrice de validation	80

Liste des exemples

3.1.1	Le graphe de régulation de <i>Pseudomonas aeruginosa</i>	23
3.1.2	les paramètres associés au graphe de régulation de <i>Pseudomonas aeruginosa</i>	24
3.1.3	Le réseau de régulation de <i>Pseudomonas aeruginosa</i>	25
3.2.1	Nombre d'états pour le réseau de régulation de <i>Pseudomonas aeruginosa</i>	25
3.2.2	états d'application des paramètres de <i>Pseudomonas aeruginosa</i>	27
3.2.3	Cas d'un paramètres non opérable	28
3.2.4	Une paramétrisation possible pour <i>Pseudomonas aeruginosa</i>	29
3.2.5	Création du graphe de transitions pour <i>Pseudomonas aeruginosa</i>	30
3.2.6	Les boucles positives	31
3.2.7	Boucle négatives	32
5.1.1	Graphe de régulation de <i>Pseudomonas aeruginosa</i> avec variable d'environnement	50
5.2.1	Les marqueurs de <i>Pseudomonas aeruginosa</i>	51
5.2.2	Les environnements pour <i>Pseudomonas aeruginosa</i>	51
5.2.3	Propriété environnementale pour <i>Pseudomonas aeruginosa</i>	52
5.3.1	Graphe de transitions possible pour un état stationnaire	52
5.3.2	Traduction formelle de l'état stationnaire de Operon chez <i>Pseudomonas aeruginosa</i>	53
5.3.3	Graphe de transitions pour un comportement oscillatoire	53
5.3.4	Le motif NR	55
6.1.1	Graphe de régulation local pour le paramètre $K_{MucB,prod}$	60
6.1.2	expérience de pensée pour $K_{MucB,prod}$	60
6.2.1	Fichier d'entrée de <i>TotemBioNet</i> pour le réseau de <i>Pseudomonas aeruginosa</i>	63
7.1.1	Transition entre deux plan <i>via</i> la variable Calcium	68
7.1.2	Les autorégulations dans un cas théorique	69
7.1.3	Séparation des plans <i>via</i> les artéfacts	69
7.1.4	Séparation des plans <i>via</i> les artéfacts chez <i>Pseudomonas aeruginosa</i>	70
7.1.5	Formules associés à la méthode de vérification globale pour <i>Pseudomonas aeruginosa</i>	72
7.1.6	Performances technique pour le modèle <i>Pseudomonas aeruginosa</i> avec la méthode globale	72
7.2.1	Nombre d'états pour le graphe global VS graphe dans un environnement	73
7.2.2	Paramètres non opérables dans les différents environnements pour <i>Pseudomonas aeruginosa</i>	74
7.2.3	Graphe de transitions pour l'environnement e^0 et la paramétrisation P	74
7.2.4	Abstraction de modèles d'un environnement	76
7.2.5	Performances techniques pour la vérification environnement par environnement du contexte de modélisation de <i>Pseudomonas aeruginosa</i>	79
8.1.1	Ajout d'un multiplexe au réseau de régulation de <i>Pseudomonas aeruginosa</i>	88

8.1.2 Paramètres pour l'ajout d'un multiplexe positif pour <i>Pseudomonas aeruginosa</i>	89
8.1.3 Paramètres pour l'ajout d'un multiplexe négatif pour <i>Pseudomonas aeruginosa</i>	89
8.1.4 Ajout théorique d'un multiplexe sur une variable d'environnement	90
8.2.1 Graphe de régulation enrichi avec <i>Calcium</i>	92
8.2.2 Paramètres pour l'ajout de la <i>Caclium</i>	93
8.2.3 Matrice pour l'ajout de <i>Calcium</i>	93
8.3.1 Réseau booléen de <i>Pseudomonas aeruginosa</i>	95
8.3.2 Formalisation du dédoublement de <i>operon</i>	96
8.3.3 Paramètres pour l'ajout d'un niveau à <i>operon</i>	96
8.3.4 Matrice de validation pour l'ajout d'un niveau à <i>operon</i>	97
8.4.1 Contexte biologique pour la scission de <i>operon</i>	99
8.4.2 Formalisation de la scission de la variable <i>operon</i>	99
8.4.3 Assignation des régulations en aval de la variable <i>operon</i>	99
8.4.4 Analyse des régulations en aval de la variable <i>operon</i>	100
8.4.5 Enrichissement partiel du graphe de régulation pour la scission de la variable <i>operon</i>	101
8.4.6 Assignation des régulations en amont de la variable <i>operon</i>	101
8.4.7 Formalisation graphique du lien entre les variables filles <i>operon</i> et <i>AlgU</i>	102
8.4.8 Graphe de régulation enrichis après scission de la variable <i>operon</i>	102
8.4.9 Réduction du nombre de paramètres pour la scission d'une variable	103
8.4.10 Association paramètres de la variable mère aux variables filles	103
8.4.11 Paramètres des variables en aval de la scission de la variable <i>operon</i> dans <i>Pseu-</i> <i>domonas aeruginosa</i>	104
8.4.12 Matrice de validation pour la scission de la variable <i>operon</i>	104

Annexes

ANNEXES **A**

Application sur le modèle de la régulation du métabolisme

A Notations pour le modèle de la régulation du métabolisme

variables

<i>AA</i>	Entrée d'acides aminés
<i>ATP</i>	Ratio entre ATP/ADP+AMP
<i>FA</i>	Entrée d'acides gras (Fatty acids)
<i>FERM</i>	Fermentation
<i>GLC</i>	Entrée de sucres sous forme de glucose
<i>GLYC</i>	Glycolyse
<i>InO2</i>	Entrée d'oxygène
<i>KRAS</i>	Mutation sur le gène KRAS aboutissant au cancer
<i>KREBS</i>	Cycle de Krebs
<i>LBP</i>	Production de biomasse lipidique
<i>METdenovo</i>	méthionine produite <i>de novo</i> via dérégulation cancéreuse
<i>NADH</i>	Ratio des cofacteurs connus NAD(p)H/NAD(p)H et FADH/FAD
<i>NCD</i>	Donneurs d'azote et carbone, acides aminés décomposés
<i>NLBP</i>	Production biomasse non lipidique
<i>O2</i>	Oxygène
<i>PHOX</i>	Phosphorylation oxidative
<i>ROS</i>	Espèces réactives à l'oxygène
<i>TGF - β</i>	facteur de transcription bêta

A.1 Multiplexes

<i>AAS</i>	Synthèse d'acides aminés
<i>AA1</i>	Entrée normale d'acides aminés
<i>AA2</i>	Forte entrée d'acide aminés
<i>A – KG</i>	Multiplexe indiquant la régulation du cycle de Krebs par l'alpha-cétoglutarate
<i>AnO</i>	Acétyl-coA et oxygène présents
<i>Box</i>	Voie métabolique de la β -oxydation
<i>COF</i>	Cofacteurs de la glycolyse
<i>GLC1</i>	Entrée normale de sucres (glucose)
<i>GLC2</i>	Forte entrée de sucres (glucose)
<i>GLN_{metabolism}</i>	Métabolisme cancéreux de la glutamine
<i>GSHc</i>	Réducteurs de ROS (glutathion) dans le PDAC
<i>LS</i>	Synthèse de lipides
<i>NOX</i>	Producteurs de ROS (NOX) dans le PDAC
<i>PC</i>	Contrôle de la phosphorylation oxydative
<i>PPP</i>	Voie des pentoses phosphates

B Liste des paramètres du modèle de la régulation du métabolisme pour une cellule cancéreuse agressive.

# Paramètres del'ATP	# Paramètres de GLYC	$K_{NADH, GLNm FERM} = 1$
$K_{ATP} = 0$	$K_{GLYC} = 0$	$K_{NADH, GLNm PHOX} = 1$
$K_{ATP, LBP} = 0$	$K_{GLYC, GR} = 0$	$K_{NADH, GLNm GLYC PHOX} = 1$
$K_{ATP, nLBP} = 0$	$K_{GLYC, GLC1} = 0$	$K_{NADH, GLNm FERM GLYC} = 1$
$K_{ATP, LBP nLBP} = 0$	$K_{GLYC, GLC1 GR} = 0$	$K_{NADH, GLNm FERM PHOX} = 1$
$K_{ATP, PHOX} = 1$	$K_{GLYC, GLC1 GLC2} = 0$	$K_{NADH, GLNm FERM KREBS} = 1$
$K_{ATP, nLBP PHOX} = 1$	$K_{GLYC, GLC1 GLC2 GR} = 0$	$K_{NADH, GLNm GLYC KREBS} = 1$
$K_{ATP, LBP nLBP PHOX} = 2$	$K_{GLYC, COF} = 0$	$K_{NADH, GLNm KREBS PHOX} = 1$
$K_{ATP, GLYC1} = 0$	$K_{GLYC, COF GR} = 0$	$K_{NADH, GLNm FERM GLYC KREBS} = 1$
$K_{ATP, GLYC1 nLBP} = 0$	$K_{GLYC, COF GLC1} = 0$	$K_{NADH, GLNm FERM GLYC PHOX} = 2$
$K_{ATP, GLYC1 LBP nLBP} = 2$	$K_{GLYC, COF GLC1 GR} = 1$	$K_{NADH, GLNm GLNm GLYC KREBS PHOX} = 1$
$K_{ATP, GLYC1 GLYC2} = 1$	$K_{GLYC, COF GLC1 GLC2} = 1$	$K_{NADH, GLNm FERM KREBS PHOX} = 1$
$K_{ATP, GLYC1 GLYC2 nLBP} = 1$	$K_{GLYC, COF GLC1 GLC2 GR} = 2$	$K_{NADH, GLNm FERM GLYC KREBS PHOX} = 2$
$K_{ATP, GLYC1 GLYC2 LBP nLBP} = 2$	# Paramètres de nLBP	$K_{NADH, GLNm AAS} = 1$
$K_{ATP, GLYC1 PHOX} = 1$	$K_{nLBP} = 0$	$K_{NADH, GLNm GLYC AAS} = 1$
$K_{ATP, GLYC1 nLBP PHOX} = 1$	$K_{nLBP, PPP} = 1$	$K_{NADH, GLNm KREBS AAS} = 1$
$K_{ATP, GLYC1 LBP nLBP PHOX} = 2$	$K_{nLBP, AAS} = 1$	$K_{NADH, GLNm FERMAAS} = 1$
$K_{ATP, GLYC1 GLYC2 PHOX} = 1$	$K_{nLBP, AAS PPP} = 1$	$K_{NADH, GLNm AAS PHOX} = 1$
$K_{ATP, GLYC1 GLYC2 nLBP PHOX} = 1$	# paramètres de NADH	$K_{NADH, GLNm GLYC AAS PHOX} = 1$
$K_{ATP, GLYC1 GLYC2 LBP nLBP PHOX} = 2$	$K_{NADH} = 0$	$K_{NADH, GLNm FERM AAS PHOX} = 1$
# Paramètres de LBP	$K_{NADH, FERM} = 0$	$K_{NADH, GLNm FERM KREBS AAS} = 1$
$K_{LBP} = 0$	$K_{NADH, FERM} = 0$	$K_{NADH, GLNm GLYC KREBS AAS} = 1$
$K_{LBP, LS} = 1$	$K_{NADH, PHOX} = 0$	$K_{NADH, GLNm KREBS AAS PHOX} = 2$
$K_{LBP, BOX} = 0$	$K_{NADH, AAS} = 0$	$K_{NADH, GLNm FERM GLYC KREBS AAS} = 2$
$K_{LBP, LS BOX} = 1$	$K_{NADH, FERM PHOX} = 0$	$K_{NADH, GLNm FERM GLYC AAS PHOX} = 2$
# Paramètres de NCD	$K_{NADH, FERMAAS} = 0$	$K_{NADH, GLNm GLYC KREBS AAS PHOX} = 2$
$K_{NCD} = 0$	$K_{NADH, FERMAAS PHOX} = 0$	$K_{NADH, GLNm FERM KREBS AAS PHOX} = 1$
$K_{NCD, Krebs} = 0$	$K_{NADH, FERM KREBS AAS} = 0$	$K_{NADH, GLNm FERM GLYC KREBS AAS PHOX} = 2$
$K_{NCD, AA1} = 1$	$K_{NADH, GLYC AAS PHOX} = 0$	$K_{NADH, GLNm FERM KREBS AAS PHOX} = 1$
$K_{NCD, AA1 Krebs} = 1$	$K_{NADH, GLYC FERMAAS PHOX} = 2$	$K_{NADH, GLNm FERM GLYC KREBS AAS PHOX} = 2$
$K_{NCD, AA1 AA2} = 2$	$K_{NADH, Krebs FERMAAS PHOX} = 2$	$K_{NADH, GLNm FERM KREBS AAS PHOX} = 1$
$K_{NCD, AA1 AA2 Krebs} = 2$	$K_{NADH, GLYC Krebs FERMAAS PHOX} = 2$	$K_{NADH, GLNm FERM GLYC KREBS AAS PHOX} = 2$
# Paramètres de PHOX	$K_{NADH, GLYC Krebs AAS} = 0$	$K_{NADH, GLNm FERM KREBS AAS PHOX} = 1$
$K_{PHOX} = 0$	$K_{NADH, FERM GLYC AAS} = 0$	$K_{NADH, GLNm FERM GLYC KREBS AAS PHOX} = 2$
$K_{PHOX, PC} = 1$	$K_{NADH, GLYC Krebs} = 0$	$K_{NADH, GLNm FERM GLYC KREBS AAS PHOX} = 2$
# Paramètres de FERM	$K_{NADH, FERM KREBS} = 0$	# Paramètres de ROS
$K_{FERM} = 0$	$K_{NADH, Krebs AAS} = 0$	$K_{ROS} = 0$
$K_{FERM, EP} = 1$	$K_{NADH, GLYC AAS} = 0$	$K_{ROS, NOX} = 1$
# Paramètres de Krebs	$K_{NADH, GLYC PHOX} = 0$	$K_{ROS, GSHa NOX} = 2$
$K_{Krebs} = 0$	$K_{NADH, FERM GLYC} = 0$	$K_{ROS, GSHa GSHc NOX} = 3$
$K_{Krebs, AnO} = 1$	$K_{NADH, Krebs} = 0$	# Paramètres de MET de Novo
$K_{Krebs, AnO a-KG} = 2$	$K_{NADH, GLYC} = 0$	$K_{METdeNovo} = 0$
$K_{Krebs, BOX} = 1$	$K_{NADH, FERM GLYC Krebs AAS} = 2$	$K_{METdeNovo, EMT} = 1$
# Paramètres de l'O2	$K_{NADH, Krebs PHOX} = 2$	
$K_{O2} = 0$	$K_{NADH, Krebs FERM PHOX} = 2$	
$K_{O2, PHOX} = 0$	$K_{NADH, Krebs AAS PHOX} = 2$	
$K_{O2, exO2} = 1$	$K_{NADH, GLYC Krebs PHOX} = 2$	
$K_{O2, exO2 PHOX} = 1$	$K_{NADH, FERM GLYC Krebs PHOX} = 2$	
	$K_{NADH, GLYC Krebs AAS PHOX} = 2$	
	$K_{NADH, GLNm} = 0$	
	$K_{NADH, GLNm GLYC} = 0$	
	$K_{NADH, GLNm KREBS} = 0$	

Table A.1 – Valeurs des paramètres du modèle de la régulation du métabolisme pour une cellule cancéreuse dans le pancréas agressif ou non. 10 valeurs des paramètres restent libres : $K_{ATP, LBP PHOX}$, $K_{ATP, GLYC1 LBP}$, $K_{ATP, GLYC1 GLYC2 LBP}$, $K_{ATP, GLYC1 LBP PHOX}$, $K_{ATP, GLYC1 GLYC2 LBP PHOX}$, $K_{NADH, FERM GLYC PHOX}$, $K_{NADH, GLNm FERM GLYC PHOX}$, $K_{NADH, FERM GLYC Krebs}$, $K_{NADH, GLNm FERM GLYC Krebs}$, $K_{ROS, GSHc}$, $K_{ROS, GSHc NOX}$, $K_{ROS, GSHa}$, $K_{ROS, GSHa GSHc}$.

Cycle de vie des modèles discrets et application à la régulation du métabolisme des cellules cancéreuses du pancréas

Laetitia GIBART

Résumé

La modélisation des systèmes biologiques complexes ne bénéficie pas des méthodes et outils existants pour le développement des produits technologiques, comme en génie logiciel. La raison principale en est l'impossibilité de mettre en place une conception modulaire : en biologie, l'introduction d'un nouvel élément dans un modèle préalablement validé peut drastiquement remettre en cause son comportement. Cette thèse fournit néanmoins une méthode incrémentale de conception de modèles discrets, pour la biologie, qui possède certaines similarités avec les méthodes de développement de logiciel en spirale. L'objectif principal de l'approche est de préserver la faisabilité des preuves permettant de vérifier la non-régression à chaque enrichissement. Bien que la méthode proposée ait été développée dans le cadre du formalisme de René Thomas, elle peut s'adapter à tout autre cadre de modélisation discret. Elle comprend trois phases distinctes : l'analyse des besoins, l'enrichissement du modèle et sa validation par des méthodes formelles et enfin une phase de prédiction. L'étude de cas, conséquente, qui ancre ce travail, est un modèle de régulation du métabolisme à différents stades d'évolution de l'adénocarcinome ductal du pancréas (PDAC). Nous adoptons dans cette thèse une approche de modélisation abstraite en partant d'un modèle général, déjà conséquent, de la régulation du métabolisme eucaryote, que nous enrichissons pas à pas pour atteindre notre objectif : un modèle couvrant l'évolution de la cellule saine à la cellule agressive en passant par la cellule cancéreuse non agressive. On ne dispose a priori que de peu de connaissances moléculaires sur les dérégulations du métabolisme mises en jeu et le niveau d'abstraction de ce modèle est choisi pour orienter au mieux les biologistes dans leurs recherches et dans le choix de leurs hypothèses de travail.

Mots-clés : Conception de modèles complexes, Modélisation discrète, Réseaux de régulation biologique, Réseaux de R. Thomas, Cancer du pancréas, bio-informatique.

Abstract

The modelling of complex biological systems does not benefit from the methods and tools that exist for the development of technological products in software engineering. The main reason for this is the impossibility of implementing a modular design: in biology, the introduction of a new element in a previously validated model can drastically change its behaviour. Nevertheless, this thesis provides an incremental methodology, in order to design discrete models for biology, which has some similarities to software spiral development model. The principal goal is to preserve the feasibility of proofs allowing to verify the non-regression at each enrichment. Although the proposed method was developed within the framework of René Thomas' formalism, it can be adapted to any other discrete modelling framework. It includes three distinct phases: requirements analysis, model enrichment combined with validation by formal methods and finally a prediction phase. The substantial case study that anchors this work is a model of metabolic regulation at different stages of progression in pancreatic ductal adenocarcinoma (PDAC). In this thesis, we adopt an abstract modelling incremental approach, whose starting point is an already consequent general model of eukaryotic metabolic regulation. Our method allows us to enrich this initial model step by step to reach our aim, that is, a model covering the evolution from the healthy cell to the aggressive cell through the non-aggressive cancer cell. One has only little molecular *a priori* knowledge on the metabolic deregulation involved, and the level of abstraction of this model is chosen to best guide biologists in their research and in the choice of their working hypotheses.

Keywords: Complex model design, Discrete modelling, Biological regulatory networks, Thomas' regulatory networks, Pancreatic cancer, bioinformatics.