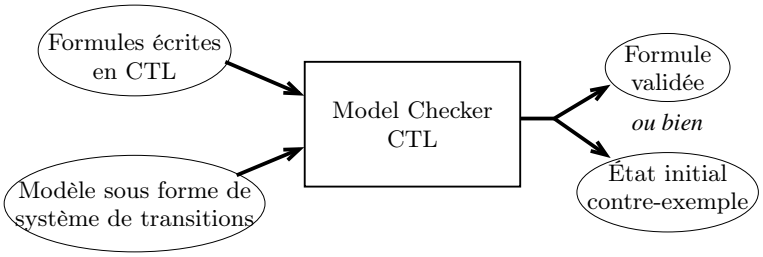
Cours n°5

1 La logique CTL en pratique

Ce qui fait la force de la logique CTL, c'est surtout l'existence d'algorithmes de *Model Checking* extrêmement rapides. Ces algorithmes prennent en entrée un automate et une formule CTL, et déterminent très rapidement si la formule est vraie à partir de tous les états initiaux ou non. Si elle ne l'est pas, la plupart de ces algorithmes fournissent un état initial contre-exemple, où la formule est fausse.



Il résulte de tout ceci que CTL et son *Model Checking* permettent, de manière très efficace, d'établir des propriétés sur les modèles de systèmes biologiques à base d'automate. Ceci sans avoir besoin d'effectuer des quantités énormes de simulations pour ce convaincre de la véracité de ces propriétés.

Exercice 1 : On a traité un exemple en HSIM avec 2 gènes `g1` et `g2` produisant mutuellement le facteur de transcription de l'autre. On a constaté que ce système, avec des paramètres appropriés, possédait deux comportements asymptotiques possibles : soit `g1` et `g2` tendaient tous deux vers un niveau d'expression faible, soit ils tendaient tous deux vers un niveau d'expression fort. Exprimez en CTL cette propriété (existence de deux bassins d'attraction de la dynamique du système).

Exercice 2 : De même pour la glycolyse, on a constaté qu'en investissant initialement un niveau moyen d'`ATP`, il devient faible pour démarrer la glycolyse, puis passe à un fort niveau. Exprimez cette propriété en CTL.

Exercice 3 : Lors du TD du cours précédent sur FBA, on a constaté que le métabolite interne c disparaissait. Exprimez cette propriété en CTL.
Pouvez-vous écrire la propriété indiquant que si l'on ne fournit pas de métabolite externe a alors ultérieurement on ne produira plus de f ?

2 BIOCHAM booléen, la syntaxe

BIOCHAM est un langage à base de règles mais sans aucune prise en compte de l'espace. Il gère un ensemble d'espèces biochimiques sans distinction de compartiments et en faisant *a priori* l'hypothèse d'une répartition uniforme. De plus, dans sa version booléenne, BIOCHAM ne considère que la présence ou l'absence de chaque espèce biochimique. Ces simplifications ont principalement deux avantages : avoir des simulations extrêmement simples et surtout disposer d'une capacité de preuves automatiques de propriétés en utilisant la logique CTL. C'est donc un choix inverse de celui des SMA : on s'abstrait des détails pour gagner en capacité à établir formellement des propriétés comportementales.

Le nom « BIOCHAM » est issu d'une théorie préexistante appelée CHAM pour CHimical Abstract Machine. Cette théorie permettait simplement de modéliser des réactions chimiques et BIOCHAM a complété son langage en le spécialisant aux espèces biochimiques dans le cadre des réseaux de signalisation, et en apportant par la suite de nombreuses méthodes informatiques pour analyser les modèles obtenus par ces systèmes de règles et leur dynamique. Pour ce faire, BIOCHAM est un environnement informatique fondé sur une *syntaxe* unique mais que l'on peut interpréter selon plusieurs *sémantiques* : on en verra deux dans ce cours. Mais revenons à la syntaxe :

2.1 Les objets

Un objet en BIOCHAM est simplement un *identifiant*, avec plusieurs conventions de notation afin de faciliter la compréhension « biochimique » de chaque objet :

- Un objet qui ne contient que des lettres et des chiffres représente une molécule « simple ».
- Un objet avec un « - » représente la formation d'un complexe. Par exemple PER1-CRY1 est une molécule qui représente le complexe obtenu à partir de PER1 et CRY1. Le « - » est supposé commutatif et associatif, c'est à dire que a-b-c représente le même objet que b-c-a ou c-b-a, *etc.* et que former d'abord le complexe a-b puis le complexe a-b-c donne le même objet que de former d'abord b-c puis a-b-c. Pour rappel : en Hsim, $*$ n'est que commutatif, pas associatif.
- On peut distinguer des « sites » sur une molécule (par exemple des sites de phosphorylation). Par exemple m∼{s1,s2,s3} représente une molécule dont les trois sites s1, s2 et s3 sont occupés.
- Enfin on peut considérer des objets abstraits qui ne sont pas des molécules avec des noms d'objet qui commencent par « @ », par exemple @DepletionCalcique.

Il est possible de déclarer à l'avance l'ensemble des sites d'intérêt d'une molécule donnée, ce qui contraint alors la forme correspondante m∼{s1,s2,s3}. La déclaration se fait sous la forme :

declare m∼{E1,...,En}

où les Ei sont des *ensembles* de sites. Par exemple la déclaration MEK∼{{},{p1},{p2},{p1,p2}} indique que MEK peut avoir 2 sites de phosphorylation p1 et p2, et que tous les cas de figure sont autorisés (aucun des sites n'est phosphorylé, l'un, l'autre ou les deux).
Lorsque tous les sous-ensembles d'un ensemble de site sont autorisés, on peut simplifier la déclaration avec

declare m∼parts-of(E)

Par exemple, pour MEK, il aurait été équivalent de déclarer MEK∼parts-of({p1,p2}).

2.2 Les solutions

Une « *solution* » en BIOCHAM est un ensemble d'objets et elle est notée avec des « + ». Ainsi « o1 + o2 + o3 » est la solution qui contient les trois objets o1, o2 et o3. La solution vide est notée « _ ».

2.3 Les réactions

Une réaction est une règle de la forme

sol1 => sol2

où sol1 et sol2 sont des solutions. Intuitivement, elle traduit une réaction biochimique.

Il existe quelques abréviations utiles :
- (sol1 =[C]=> sol2) est une abréviation de (C + sol1 => C + sol2) et signifie que C catalyse la réaction (sol1 => sol2)
- (sol1 =[sol3=>sol4]=> sol2) est une abréviation de (sol3 + sol1 => sol4 + sol2) et signifie que la réaction (sol3 => sol4) est vue comme un catalyseur de la réaction (sol1 => sol2)
- (sol1 <=> sol2) est une abréviation des deux réactions (sol1 => sol2) et (sol2 => sol1).
- Enfin (sol1 <=[C]=> sol2) est une abréviation des deux réactions (sol1 =[C]=> sol2) et (sol2 =[C]=> sol1).

2.4 Les motifs

De même que certaines abréviations, les motifs (patterns) permettent d'écrire plusieurs réactions en une seule fois. Ils sont liés aux déclarations des objets en utilisant des *variables* qui sont des identifiants commençant pas un « $ ». Par exemple, avec la déclaration « declare MEK∼parts-of({p1,p2}) », la réaction suivante

MEK∼$p + RAF∼{p1} <=> MEK∼$p-RAF∼{p1}

s'expande en 4 réactions où « $p » est successivement remplacé par {}, {p1}, {p2} et {p1,p2}. Soit 8 réactions au total puisque chaque « <=> » s'expande lui-même en 2 réactions. Elle signifie simplement que, pourvu que RAF soit

phosphorylé, il peut former un complexe avec MEK quel que soit sont état de phosphorylation, et ce complexe peut se dissocier.

Nota : ce serait une erreur d'écrire (MEK + RAF~{p1} <=> MEK-RAF~{p1}) car « MEK » seul ne fait pas partie de la déclaration « MEK~parts-of({p1,p2}) » et MEK seul est donc mal formé.

Enfin des *contraintes* peuvent être données sur les variables avec le mot clef « where » et le prédicat d'appartenance « in ». Par exemple :

MEK~$p + RAF~{p1} <=> MEK~$p-RAF~{p1} ~ where p2 not in $p

restreint la réaction aux ensembles $p qui ne contiennent pas p2 (donc {} et {p1}), i.e. aux cas où MEK n'est pas déjà phosphorylé sur le site p2.

3 BIOCHAM booléen, la sémantique

L'idée est que, partant d'un état qui est représenté par une solution (l'ensemble des objets présents à un instant donné), on applique une réaction qui modifie la solution, puis on recommence tant que c'est possible.

3.1 Réactions applicables

On remarque tout d'abord qu'étant donné un état de départ, toutes les réactions ne sont pas applicables. Il faut évidemment que tous les objets du côté gauche de la réaction soient présents dans la solution de départ.

Étant donné une solution globale S, une réaction (sol1 => sol2) est *applicable* si et seulement si sol1 $\subseteq$ S.

3.2 Applications possibles d'une réaction

Si la réaction (sol1 => sol2) est applicable en partant de la solution courante S, il est évident que lorsqu'on l'applique, on obtient une solution S' où chaque objet de sol2 est présent.

Pour chaque objet o de sol1, c'est plus subtil car on sait qu'il est présent dans S (sinon la réaction ne serait pas applicable) mais on ne sait pas en quelle quantité. Par conséquent il est possible qu'en appliquant la réaction (sol1 => sol2) on épuise toutes les « réserves » de l'objet o, *ou pas*. Bien sûr si l'objet o est dans sol2, il va se retrouver dans S', mais s'il est dans sol1 mais pas dans sol2, il faut considérer les deux possibilités.

Enfin, tous les objets qui ne sont pas dans sol1 restent inchangés : ils sont dans S' si et seulement si il étaient déjà dans S.

Ainsi, appliquer la réaction

a + b + c => c + d

à partir d'une solution initiale S={a,b,c,e} peut fournir l'une des quatre solutions S'={a,b,c,d,e} (rien n'a été épuisé), S'={a,c,d,e} (b a été épuisé), S'={b,c,d,e} (a a été épuisé) ou S'={c,d,e} (a et b ont été épuisés).

Lorsqu'un système de plusieurs réactions est considéré à partir d'un état S, on doit donc d'abord faire l'inventaire de toutes les règles applicables, puis pour chacune d'elles considérer toutes les possibilités précédentes. On obtient alors un *ensemble de successeurs possibles* de l'état S et le choix est non déterministe.

Etant donné un système de réactions, on peut faire l'inventaire E de tous les objets qui sont possiblement consommés ou produits par ces règles. Cet ensemble E donne lieu à autant d'états initiaux qu'il a de sous-ensembles (soit 2^n états à considérer si n est le nombre d'éléments de E). Pour chacun de ces états, on inventorie les réactions applicables et toutes leurs applications possibles, et cela construit le graphe de transitions (encore appelé graphe d'états) qui décrit la dynamique du système. On peut alors y chercher quels sont les états stables, quels sont les chemins possibles, si un état donné est atteignable en partant d'un autre, *etc.*

4 TD : Formation de complexes en BIOCHAM

Dans cette partie du TD, on n'utilise que la construction de la syntaxe de BIOCHAM permettant de faire des complexes à partir de molécules élémentaires *via* le symbole « − » (les sites, gènes et propriétés sont ignorés). On se focalise de plus sur une réaction simple où une protéine E joue le rôle d'enzyme pour transformer un substrat s_1 en s_2. Plus précisément E a 1 site ayant de l'affinité pour 1 molécule s_1 et relargue 1 molécule de s_2.

Exercice 4 : Écrivez en BIOCHAM la réaction qui traduit cette information sur E, s_1 et s_2.

On suppose à partir de maintenant que E est un complexe formé de deux molécules a, que s_1 est un complexe formé de trois molécules b et que s_2 est un complexe formé de deux molécules b.

Exercice 5 : Quel ensemble minimum de molécules élémentaires $\mathcal{M}$ faut-il considérer pour engendrer avec l'opération « − » de BIOCHAM un ensemble de complexes contenant E, s_1 et s_2 ? Décrivez l'ensemble de tous les complexes engendrés par $\mathcal{M}$.

5 TD : Sémantique booléenne de BIOCHAM

Exercice 6 : Récrivez la réaction de l'exercice 4 en remplaçant les symboles E, s_1 et s_2 par leur définition en tant que complexes. Cette réaction n'est pas intuitive en termes de conservation de la matière ; pourquoi ?

Dans la suite on considère le système des trois réactions de réaction suivantes :

1. $a\text{-}a\text{-}a\text{-}a + a\text{-}a \Longrightarrow a\text{-}a\text{-}a\text{-}a + a$
2. $a\text{-}a\text{-}a\text{-}a + a\text{-}a \Longrightarrow a\text{-}a$
3. $a + a\text{-}a \Longrightarrow a$

que l'on écrira de manière simplifiée :

1. $a^4 + a^2 \Longrightarrow a^4 + a$
2. $a^4 + a^2 \Longrightarrow a^2$
3. $a + a^2 \Longrightarrow a$

Exercice 7 : Chacune de ces trois réactions préserve-t-elle le « principe de conservation de la matière » ; pourquoi ?

Exercice 8 : Quel ensemble minimum de molécules élémentaires faut-il considérer pour engendrer, toujours avec « − », un ensemble de complexes contenant ceux des réactions (1.) à (3.) ?

Exercice 9 : Quelles sont les solutions utiles pour tracer le graphe d'états associé aux réactions (1.) à (3.) en supposant qu'on ne part que d'état initiaux où aucune molécule ne reste toujours inerte ?

Exercice 10 : Partant d'une solution qui ne contient aucun complexe a^4, est-il possible d'atteindre une solution qui en contienne ? Pourquoi ?

Exercice 11 : Partant d'une solution qui ne contient que des complexes a^4, est-il possible d'atteindre une solution qui n'en contienne plus ? Pourquoi ?

Exercice 12 : Partant d'une solution qui ne contient aucun complexe a^2, est-il possible d'atteindre une solution qui en contienne ? Pourquoi ?

Exercice 13 : Tracez le graphe d'états associé aux réactions (1.) à (3.).

Exercice 14 : Partant d'une solution qui contient à la fois des complexes a^4 et a^2 :
- est-il possible d'atteindre une solution qui n'en contienne plus ? (ni a^4, ni a^2)
- est-il possible d'atteindre une solution vide (aucune molécule) ?

Argumentez en utilisant le graphe de la question 13.

Exercice 15 : Traduisez les deux questions de l'exercice précédent en CTL, puis dites pour chacune des formules si elle est valide dans le modèle BIOCHAM considéré.

Ainsi, la version booléenne de BIOCHAM est idéale pour faire des preuves automatiques de comportements invariants du système. Elle en apporte une description *qualitative* qui suffit à étudier les principales propriétés du système. Ceci de manière automatique grâce au model checking de CTL.

6 Sémantique ODE de BIOCHAM

En terme de simulation, la sémantique booléenne est une peu pauvre. En effet, partant d'un état initial donné, on en est réduit, à chaque étape, à tirer aléatoirement quelle réaction applicable on va appliquer, puis pour chaque objet du membre gauche de la règle, tirer aléatoirement s'il sera « épuisé » ou non. Cela revient à construire aléatoirement un chemin dans le graphe de transitions.

Il existe une version *quantitative* de BIOCHAM, qui bien sûr, perd en capacité de preuves assistées par ordinateur, mais donne lieu à des simulations plus précises. Pour cela, il faut essentiellement compléter la syntaxe par deux choses : la stœchiométrie des réactions et leur vitesse.

6.1 Solutions avec stœchiométrie *vs.* état du système biochimique

Au niveau des solutions, la différence avec le cadre booléen, c'est qu'une solution est un *multi-ensemble* d'objets. Par exemple « $o_1 + 3 * o_2 + 4 * o_3$ » est la solution qui contient 1 exemplaire de o_1, 3 exemplaires de o_2 et 4 exemplaires de o_3. La forme générale d'une solution est donc « $n_1 * o_1 + n_2 * o_2 + \cdots$ ».

Par ailleurs, l'état d'un système à un instant donné n'est plus une simple solution comme dans le cas booléen : c'est la donnée, pour chaque objet, de son *taux de concentration*.

6.2 Réactions avec cinétique

Enfin on munit chaque réaction d'une *cinétique*, qui est une expression e permettant de calculer la vitesse de la réaction en fonction de ses réactants. La forme générale d'une réaction devient alors :

$$e \quad for \quad g_1 * o_1 + \cdots + g_m * o_m \implies d_1 * o'_1 + \cdots + d_n * o'_n$$

où les g_i et d_i sont les coefficients de stœchiométrie (gauche et droite) et les o_i et o'_i les objets de la réaction.

Les deux cinétiques les plus courantes sont la loi d'action de masse et Michaelis-Menten.

- La syntaxe des réactions selon la loi d'action de masse est :
$$MA(k) \quad for \quad g_1 * o_1 + \cdots + g_m * o_m \implies d_1 * o'_1 + \cdots + d_n * o'_n$$
et elle signifie que la vitesse de la réaction est $k \times o_1^{g_1} \times \cdots \times o_m^{g_m}$ (la fameuse constante d'équilibre de la loi d'action de masse, en considérant que o_i dénote le taux de concentration de l'objet o_i).
Lorsque la réaction est réversible, ce qui est presque toujours le cas, on regroupe les deux sens de la réaction comme suit :
$$MA(k, k') \quad for \quad g_1 * o_1 + \cdots + g_m * o_m \iff d_1 * o'_1 + \cdots + d_n * o'_n$$
k' est alors la constante pour la réaction de droite à gauche.
- La syntaxe des réactions selon Michaelis-Menten (où C est l'enzyme, o le substrat et o' le produit) devient :
$$MM(V, K) \quad for \quad o \ = [C] => \ o'$$
et elle signifie que la vitesse de la réaction est $\frac{V \times o}{K + o}$, où V peut être calculé à partir d'un paramètre k avec $V = k \times \frac{C + C \times o}{K}$.

En fait tout calcul de vitesse de réaction e est autorisé par la syntaxe générale. Par exemple on peut imposer des fonctions de Hill avec :

$$(V * o^n)/(K^n + o^n) \quad for \quad o \ = [C] => \ o'$$

donc e est la fraction $\frac{V \times o^n}{K^n + o^n}$, toujours avec $V = k \times \frac{C + C \times o}{K}$.
En fait, $MM(V, K)$ est une fonction de Hill où $n = 1$.
Nota : lorsque n est plus grand, la fonction de Hill se rapproche d'une fonction en escalier et, bien souvent, une modélisation discrète est alors préférable.

6.3 Simulations par équations différentielles

Lorsqu'on a écrit un ensemble de réactions avec des cinétiques, les paramètres de ces cinétiques on généralement été estimés de manière très approximative. Ils sont souvent déterminés à partir de constantes mesurées *in vitro* qui peuvent fortement différer de la réalité *in vivo*, sans même parler des compartiments cellulaires qui contredisent l'hypothèse

d'homogénéité. Le moyen le plus direct pour mieux calibrer ces paramètres est alors la simulation par des équations différentielles ordinaires (ODE).

On doit donc traduire en ODE un système de réactions de la forme

$$\begin{array}{llll} e_1 & for & G_1 & \Longrightarrow D_1 \\ e_2 & for & G_2 & \Longrightarrow D_2 \\ \cdots & & & \\ e_p & for & G_p & \Longrightarrow D_p \end{array}$$

p est le nombre de réactions et où les G_i sont les solutions de gauche de chaque réaction et les D_i celles de droite.

Étant donné un objet o, on va noter $g_o(i)$ sa stœchiométrie dans G_i. En particulier, si o n'apparaît pas dans G_i, alors $g_o(i) = 0$. De la même façon, on note $d_o(i)$ sa stœchiométrie dans D_i.

À chaque instant, la vitesse de variation du taux de concentration de l'objet o est assez naturellement la somme des vitesses induites par chacune des p réactions. Si o apparaît dans G_i alors il est consommé à la vitesse e_i que multiplie sa stœchiométrie $g_o(i)$. Et symétriquement, si o apparaît dans D_i alors il est produit à la vitesse $e_i \times d_o(i)$. Par conséquent chaque réaction $i = 1..p$ contribue pour $e_i \times (d_o(i) - g_o(i))$ à la vitesse de o.
Au bilan, la vitesse du taux de concentration de l'objet o est donc la somme :

$$e_1 \times (d_o(1) - g_o(1)) + \ \cdots \ + e_n \times (d_o(p) - g_o(p)) = \sum_{i=1..p} (e_i \times (d_o(i) - g_o(i))).$$

Connaissant les vitesses à chaque instant des taux de concentration des objets, on suit leur taux de concentration au cours du temps en passant à l'intégrale. Malheureusement les objets ne sont pas indépendants des uns des autres car les expressions e_i font appel aux taux de concentration d'autres objets.
Pour connaître l'évolution au cours du temps de la concentration des objets du système, il faut donc résoudre un *système d'équations différentielles* contenant autant d'équations différentielles que d'objets :

$$\left\{ \begin{array}{l} \frac{do_1}{dt} = \sum_{i=1..p} e_i \times (d_{o_1}(i) - g_{o_1}(i)) \\ \frac{do_2}{dt} = \sum_{i=1..p} e_i \times (d_{o_2}(i) - g_{o_2}(i)) \\ \frac{do_3}{dt} = \sum_{i=1..p} e_i \times (d_{o_3}(i) - g_{o_3}(i)) \\ \cdots \end{array} \right.$$

Cette résolution est généralement impossible *analytiquement*, c'est-à-dire trouver une formule générale de la forme $o_i(t) = \cdots$. On ne peut que faire une simulation du systèmes d'équations différentielles, ce qu'on appelle une résolution *numérique*. Plusieurs méthodes existent (avec quelques pièges...), décrites dans de nombreux ouvrages, comme par exemple

> V. Guinot, B. Cappelaere, S.T.E. Polytech'Montpellier *Méthodes Numériques Appliquées (Résolution numérique des équations différentielles de l'ingénieur)*, Polytech'Montpellier STE 2, 2006.

... et en pratique on fait simplement appel à des logiciels pour faire ces résolutions numériques.

Les approches quantitatives par équations différentielles présentent plusieurs défauts, principalement :
- les trajectoires sont toutes *déterministes*, ce qui n'est pas crédible biologiquement
- et l'identification des paramètres s'effectue souvent « à tâtons » au travers de simulations numériques successives dans lesquelles on fait varier pas à pas les valeurs de chaque paramètre jusqu'à obtenir des courbes de variation en accord avec les connaissances biologiques ; c'est un processus long et coûteux, qui fournit seulement un jeu de paramètres crédible parmi d'autres.

Apporter des méthodes assistées par ordinateur plus exhaustives est un sujet de recherche actuel très actif. BIOCHAM est probablement l'approche la plus avancée dans le domaine de la modélisation des réseaux se signalisation, c'est pourquoi on la présente ici. Les approches à base de règles sont moins performantes pour d'autres types de réseaux, comme les réseaux de régulation typiquement, où d'autres méthodes s'avèrent plus performantes.