

1 Simulation de réseaux de régulation

On va travailler sur ce TD sur des réseaux de régulation, c'est-à-dire la simulation de l'expressivité des gènes et la concentration des protéines associés à chaque gène dans une cellule. Le but du simulateur sera donc de calculer sur un intervalle de temps donné (pour simplifier entre 0 et la durée de la simulation exprimée en secondes), la concentration de toutes les protéines associées aux gènes considérés au fil du temps. Pour simplifier, on considèrera une simulation discrète et le but sera donc de calculer seconde par seconde la concentration de chaque protéine dans une cellule. Le programme devra contenir un modèle de l'état du système et la manière de le mettre à jour après une seconde.

1.1 Simulation d'un gène

Chaque gène devra avoir un nom permettant de l'identifier de manière unique. Chaque gène X encodera une protéine $p(X)$ et sera donc responsable de la production et dégradation (et donc de sa concentration au fil du temps) de cette protéine. La production d'une protéine dépend de l'expressivité du gène X , quand X s'exprime, la cellule produit la protéine $p(X)$. On supposera que la production augmente la concentration de la protéine d'une quantité $\beta(X)$ (dépendant du gène X) correspondant à une augmentation de concentration par seconde.

Il existe également des processus qui réduisent la concentration de $p(X)$. Par exemple, la concentration de $p(X)$ peut diminuer à cause de la dilution quand la cellule grossit ou se divise ou bien parce que d'autres processus dégradent la protéine $p(X)$. La concentration $c(X)$ d'une protéine $p(X)$ décroît d'une quantité $\alpha(X) \cdot c$. Le taux $\alpha(X)$ est spécifique à la protéine $p(X)$ et donc au gène X . Le changement de concentration est proportionnel à la concentration elle-même : c'est-à-dire que nous avons une règle telle que " $c(X)$ baisse de $\alpha(X)\%$ par seconde".

1.2 Règles de régulation

Comme indiqué précédemment, la production d'une protéine d'un gène dépend de règles de régulation. Pour simplifier, on ne considèrera que des règles binaires qui indiqueront simplement si le gène s'exprime (règle active) ou non (règle inactive) et donc s'il y a une production de la protéine correspondante ou non à chaque instant. On considèrera deux types de règles : simple ou complexe. Une interaction classique entre gènes peut être représentée par des règles de promotion ou de répression. L'idée est que la protéine d'un autre gène (ou bien du gène lui-même), dite *régulatrice*, peut avoir un effet de promotion (augmentant la production de la protéine) ou de répression (diminuant la production d'un gène) sur la production de la protéine *régulée*. Une règle de promotion promeut la production de la protéine régulée (fait que le gène s'exprime) que si les deux conditions suivantes sont remplies :

- La concentration de la protéine régulatrice est supérieure ou égale à une valeur dépendant de la règle.
- Une condition appelée *signal* dépendante de la protéine régulatrice est remplie. Biologiquement, cette condition représente des stimuli extérieurs comme la présence de sucre, de lactose, ... qui va affecter le processus de transcription en modifiant la forme de la protéine régulatrice. Dans le simulateur, cette

condition appelée *signal* sera codée dans l'état du système pour chaque gène. La valeur de cette condition pourra être changé par des événements décrits ci-après.

Une règle de répression promeut la production de la protéine régulée que si au moins une des deux conditions suivantes est fausse. Parfois la régulation de la production d'une protéine ne dépend pas seulement d'une règle simple, mais de plusieurs règles de régulations. Par exemple, l'activation d'un gène peut dépendre de deux règles simples soit sous la forme d'une disjonction logique (la règle complexe promeut la production de la protéine si et seulement si au moins une des deux règles en promeut la production) ou bien d'une conjonction logique (la règle complexe promeut la production de la protéine si et seulement si les deux règles en promeut la production).

1.3 Événements

Le simulateur devra pouvoir gérer la présence d'événements. Les événements auront un temps d'activation (le moment auquel il s'active exprimé en secondes depuis le début de la simulation) et pourront modifier l'état du système d'une des deux façons suivantes :

- une mise à jour de la concentration de certaines protéines, les protéines concernées ainsi que la nouvelle concentration dépendent de l'événement ;
- une mise à jour de la condition signal de certaines protéines, les protéines concernées ainsi que la nouvelle valeur de la condition dépendent de l'événement.

2 Questions

Question 1 : Quelles sont les classes et interfaces qui vous paraissent utiles de définir pour concevoir le simulateur ?

Question 2 : Quels sont le ou les patrons de conception qui vous paraissent les plus utiles pour concevoir le simulateur ?

Question 3 : Donner un diagramme de classes du simulateur.

On souhaite maintenant rajouter une fonctionnalité d'écriture de fichier permettant de sauvegarder la configuration d'une simulation (gènes, règles et événements) à partir d'un fichier.

Question 4 : Modifier le diagramme de classes de la question précédente afin de prendre en compte l'écriture de fichier.