Sam Meyer
Projet 5BIM de simulation d'un système biologique 2019-2020

Objectif: Développer un code de simulation d'évolution permettant de tester l'adaptation de réseaux de régulation transcriptionnels en l'absence de mutations dans les séquences des gènes et des promoteurs.

Principes du modèle et de la méthode de simulation:

- La diversité évolutive est engendrée par des inversions de régions chromosomiques et de courtes insertions/délétions (une unité de longueur du code de simulation, 60 nucléotides, mais vous pourrez ensuite faire varier cette longueur)
- On simplifie la description biologique en assimilant les gènes et les transcrits, et on suppose que les débuts des promoteurs sont situés aux sites de débuts de transcription. Cela revient à identifier le début des promoteurs et le début des gènes.
- Les modifications évolutives envisagées ne peuvent toucher que des régions non-transcrites, et ainsi ne modifient ni les gènes et ni leurs promoteurs… D'après les modèles de régulation classiques, ces événements sont donc neutres, mais ce n'est pas forcément le cas dans notre système.
- Les paramètres évolutifs (taux d'apparition des deux types d'événements) seront des variables globales de votre code. En réalisant un événement à chaque génération, cela se réduit à un seul paramètre, la fréquence relative des deux événements.
- Le profil d'expression des gènes d'un individu est obtenue par simulation transcriptionnelle pendant un certain temps. Des gènes proches sont couplés par le surenroulement suivant leur orientation, distance, et niveau d'expression: les changements évolutifs ont donc un impact sur l'expression. Le code de simulation vous est fourni:
- La fitness est mesurée par la distance entre ce profil d'expression (fraction d'ARN total pour chaque gène) et un profil "cible" considéré comme parfaitement adapté à l'environnement. Il n'y a pas de choix évident pour la définition de la fitness, mais on propose le choix initial
  fitness = exp (- Σ_i | ln(f_obs_i/f_cible_i) |) où la somme porte sur les gènes
  La présence du log permet d'assurer que tous les gènes contribuent à la fitness. Vous pourrez être amenés à changer cette définition si vous vous rendez compte qu'elle n'est pas pertinente (p. ex. si un seul gène domine les valeurs de fitness ou qu'elle n'augmente pas au cours de l'évolution).
- L'évolution sera simulée par un algorithme de Monte Carlo Metropolis: (1) les événements évolutifs sont générés aléatoirement; (2) un gain de fitness est toujours accepté; une perte de fitness Δf peut être acceptée avec une probabilité exp(Δf / q), où q est un paramètre qui contrôle les amplitudes de fitness acceptables (et dont le choix est à discuter).

Contraintes:

- Codage en Python3
- Le code prend en entrée les propriétés du génome de l'individu initial (gènes, niveaux d'expression, position des barrières…) et doit calculer les génomes des générations successives au cours du temps (avec un intervalle de temps ?) en gardant la liste des événements évolutifs et la fitness. Vous tracerez un graphe avec la courbe de la fitness, en indiquant en couleur les événements évolutifs selon leur type.

Questions et rapport à rendre:

Le principe du projet est de vous laisser un maximum de liberté sur votre manière d'organiser votre code, et aussi de proposer un plan "d'expériences numériques" qui permette de répondre à l'objectif fixé plus haut. Votre rapport aura donc pour but de convaincre le lecteur que la régulation peut évoluer via des mécanismes nouveaux, non pris en compte dans les modèles usuels, et si possible de caractériser des propriétés intéressantes de ces

mécanismes. Vous êtes donc relativement libres de choisir quels génomes vous simulez, quelles analyses vous en faites etc., mais il faut que la démarche soit bien claire et logique, comme si vous écriviez un article de recherche scientifique.
Pour avoir néanmoins une partie commune, nous allons vous demander de commencer par étudier tous l'évolution du même individu, dont le génome est fourni. Ce monsieur “tousgènesidentiques” a un génome de taille 30000, constitué de 10 domaines identiques de taille 3000. Chaque domaine (bordé par deux barrières topologiques) contient un gène de taille 1000 en son centre, de même taux d'initiation 0.2 (ce taux élevé signifie en pratique que toutes les ARNPol transcrivent presque tout le temps). Les gènes ont pour noms g1, g2, …, g10. L'individu possède 10 ARNPol (mais vous pourrez ensuite le faire varier).
Monsieur tousgènesidentiques est très malheureux dans son environnement qui favorise un profil d'expression assez inhomogène, fourni dans le fichier environnement.dat.
Vous commencerez pas étudier ce système: l'évolution peut-elle résoudre le problème de Monsieur tousgènesidentiques ? L'objectif est de quantifier combien on se rapproche de l'objectif, quelles sont les modifications chromosomiques clé, etc. Vous présenterez ces résultats dans votre rapport, et vous pourrez ensuite étudier d'autres paramètres et d’autres systèmes à votre guise.

Format et contenu du rapport: max 8 pages, dont max 4 figures.
Comme annoncé plus haut, adoptez le style d’un article, avec une introduction au sujet, la question scientifique testée, une description des méthodes et choix techniques, une partie résultats et discussion. Formulez une réponse explicite à la question posée, et n’hésitez pas à indiquer les limites de votre étude.
Date limite 24 janvier minuit. Joindre le code.