Table des matières

Introduction

Chapitre 1

Analyse

1.1 Contexte

1.1.1 Sujet

Le métabolisme d'une cellule est un système complexe de transformations moléculaires et énergétiques qui se déroulent de manière ininterrompue dans la cellule et mettant en jeu un ensemble de réactions dites métaboliques. Ces réactions impliquent différents types de métabolites qui, suivant leurs positions dans la réaction, sont appelés substrats ou produits et catalysés par des enzymes.

La représentation visuelle des grandes fonctions métaboliques (glycolyse, photosynthèse ...) sous la forme de réseaux pourrait être un outil précieux pour faciliter l'avancement des travaux des chercheurs. Cette modélisation de réactions permet de réaliser des requêtes complexes comme, par exemple, le calcul (et la prédiction) de tous les métabolites pouvant être générés à partir d'un ensemble de composés sources.

Quelques logiciels disponibles internationalement permettent de travailler et d'automatiser l'étude de ces réseaux, tout en proposant des fonctionnalités telles que le calcul des modes élémentaires de flux ou la recherche de *minimal cut sets*[1]. Cependant, ils peuvent être dépendants de logiciels non libres comme MATLAB ou alors ne pas avoir d'interface utilisateur conviviale, ce qui est le cas avec *regEfmtool* (logiciel sur lequel nous nous appuierons pour ce projet).

1.1.2 Objectif

Dans le cadre de cette U.E., il nous est demandé de créer un programme (package et documentation à fournir à la fin du projet) nous permettant d'approfondir ou d'apprendre un langage de programmation, et de réaliser une analyse critique du travail effectué. L'objectif de ce projet sera donc de réaliser une interface graphique du programme *regEfmtool* qui ne peut être utilisé, à l'heure actuelle, que par des commandes à l'aide d'une console.

Deux possibilités s'offraient à nous pour réaliser cette interface : nous avions le choix entre les technologies Web et le langage Java. Nous avons choisi de nous appuyer sur les technologies Web. En effet, beaucoup de programmes utilisent une interface de type site Web leur permettant d'une part de créer une interface graphique conviviale, feuilles de style, tout en permettant une certaine modularité.

1. Un *minimal cut sets* [1] (MCS) est un ensemble minimal (irréductible)de réactions dans le réseau dont l'activation va certainement conduire à une défaillance de certaines fonctions du réseau.

1.2 État de l'existant

1.2.1 Efmtool

Efmtool [2] calcule les modes élémentaires de flux de réseaux métaboliques. Il est implémenté en Java et a été intégré à MATLAB.
Il a été développé par Marco Terzer. La version courante est la 4.7.1 (Décembre 2009).

1.2.2 *RegEfmtool*

RegEfmtool [3] est un outil informatique qui combine le calcul des modes élémentaires de flux et la régulation transcriptionnelle du réseau métabolique. Il a été développé, entre autres, par Christian Jungreuthmayer. Il a été créé afin d'accélérer le calcul de jeux complets de modes élémentaires de flux d'un réseau métabolique.
RegEfmtool est une extension d'Efmtool qui prend en compte la régulation transcriptionnelle des réseaux pour le calcul des modes élémentaires de flux.
La prise en compte de la régulation des gènes réduit de façon importante le nombre de solutions et permet d'éliminer constamment les modes qui ne peuvent exister biologiquement pendant et après le processus de calcul. Elle permet aussi de réduire considérablement le coût du calcul.
L'installation et l'utilisation de *regEfmtool* a été exclusivement testée sous Linux. Elle pourrait cependant fonctionner sous d'autres systèmes d'exploitation puisqu'il s'agit d'un programme Java. Il n'existe pas d'interface graphique de cette application, elle s'exécute donc en lignes de commandes via le terminal. La version courante de *RegEfmtool* est la 2.0 (Août 2012).

1.2.3 METATOOL

METATOOL [4] est un programme écrit en C développé de 1998 à 2000 par Thomas Pfeiffer (Berlin) en coopération avec Juan Carlos Nuno (Madrid), Stefan Schuster (Berlin) et Ferdinand Moldenhauer (Berlin).
Il sert à étudier la structure des réseaux métaboliques à partir d'équations de réactions stœchiométriques et permet notamment de calculer les modes élémentaires.
Les premières versions de METATOOL (jusqu'à la 4.9) ont été développées en C. Aujourd'hui, nous trouvons aussi une version de METATOOL en C++ mais cette version n'est pas au point. Dans la version actuelle (5.1) l'exécutable est désormais un module de MATLAB 7 et GNU 3.0 Octave, il se présente sous la forme d'un ensemble de fichiers scripts de MATLAB.

Les paramètres donnés en entrée pour le bon fonctionnement du logiciel METATOOL sont les suivants :

- La liste des réactions réversibles, ainsi que celle des réactions irréversibles, avec le nom des réactions,
- La liste des métabolites internes et externes impliqués dans les réactions,
- Les équations réactionnelles.

Le tout est rassemblé dans un fichier avec l'extension *.dat*

A la fin de son exécution, METATOOL a généré un fichier avec l'extension *.out* dans lequel se trouvent les résultats. Dans les versions de METATOOL écrites en C, le fichier de sortie contient l'ensemble des résultats sous forme de matrices, ainsi que des bilans qui permettent de décrire le réseau d'étude.
Les versions de METATOOL écrites en MATLAB produisent des résultats similaires en terme de calcul des matrices des modes élémentaires mais les résultats sont disposés différemment dans le fichier de sortie.

1.2.4 CellNetAnalyzer

CellNetAnalyzer [5] est un package de MATLAB (écrit en C) qui fourni un environnement compréhensible et convivial pour l'utilisateur et qui permet une analyse fonctionnelle et structurelle de réseaux biochimiques. Il a été développé à l'institut Max Planck de Magdeburg par Steffen Klamt (depuis 2000) et Axel von Kamp (depuis 2007) notamment.
CellNetAnalyzer fourni une importante collection d'outils et d'algorithmes pour l'analyse structurelle de réseaux.
C'est un programme gratuit pour une utilisation académique. Pour l'exécuter, il faut avoir installé MATLAB 7.0 ou une version ultérieure qui demande une licence. Il peut être utilisé sur Linux, Windows XP ou Mac.
Pour l'étude des modes élémentaires, CellNetAnalyzer fait appel à METATOOL via le logiciel MEX qui sert d'interface. MEX permet à MATLAB d'appeler tout logiciel C externe pour compléter les outils qu'il possède.

1.2.5 Yana

YANA [6] est un logiciel libre, écrit en JAVA, utilisant METATOOL pour l'étude des voies métaboliques. Il contient le logiciel METATOOL dans sa structure interne.
YANA [7] sert de façade et de sortie à METATOOL 6 tout en implémentant d'autres fonctions d'analyses du métabolisme (ex : quantification de l'activité enzymatique des réactions).
YANA s'occupe de l'entrée des données vers METATOOL puis il effectue une analyse syntaxique du fichier *.out* pour afficher les résultats sur une interface graphique et effectuer des analyses complémentaires sur ceux-ci.

1.2.6 Acom

ACoM [8] fonctionne à partir des fichiers de sortie de METATOOL. Il permet une classification automatique des modes élémentaires en fonction d'une taille minimale et d'un seuil de similarité. Il est surtout utilisé pour l'analyse des réseaux de grandes tailles où la manipulation des données à la main est laborieuse. ACoM est un programme C utilisable uniquement en mode console.

1.2.7 JACoMode

JACoMode est une interface Web permettant lancement d'ACoM via le Web. En plus d'obtenir les résultats d'ACoM, JACoMode traite ces résultats pour obtenir d'autres résultats statistiques sur les modes élémentaires.

1.2.8 Langages

Notre projet nécessite l'utilisation d'une interface Web, dans ce cadre il existe :
- Mod Perl combiné avec Apache [2] et CGI [3] mais la technologie utilisée est à l'heure actuelle dépassée
- Mod Python combiné avec Apache et CGI mais même remarque que précédemment
- CL-WHO avec Hunchentoot [4] et ParenScript offre un bon environnement pour le développement Web
- Java et ses applets [5] apparaissent également comme un bon choix pour le développement Web
- PHP [6] couplé avec du JavaScript peut être un choix judicieux pour une application Web

Parmi les différents choix précédemment cités, deux sortent du lot : Java et PHP avec leurs bibliothèques.

2. Apache HTTP Server
3. CGI : Common Gateway Interface.
4. Hunchentoot HTTP Server
5. Applet Java : logiciel s'exécutant dans la fenêtre d'une navigateur Web (grâce à une machine virtuelle Java (JVM)), fournie aux utilisateurs sous la forme de bytecode Java.
6. PHP : Hypertext Preprocessor

1.3 Analyse des besoins

1.3.1 Besoins fonctionnels

Interface Homme Machine (IHM)

L'interface Web que nous allons réaliser devra permettre de charger la description d'un réseau préexistant ou d'en créer un nouveau, mais également de modifier les descriptions de ce dernier.
Elle donnera également le moyen à l'utilisateur de saisir les fonctions et options qui l'intéressent puis de lancer les calculs des modes élémentaires de flux du réseau d'intérêt. De plus, un script récurent de commandes pourra être enregistré et chargé par la suite.

Chargement des données

Une zone de chargement de fichiers (de différents formats) à partir du disque dur de l'utilisateur sera présente sur l'interface.
Selon le type du fichier donné en paramètre d'entrée, une liste de choix possibles pourra être proposée. Ainsi les résultats d'expériences similaires (suppression, modification des réactifs, produits ou enzymes du réseau) devront pouvoir être comparés avec un affichage en vis-à-vis, ce qui implique une sauvegarde temporaire des résultats sur le serveur.

Réglage des paramètres

Il sera possible pour un utilisateur confirmé ou habitué à l'interface d'avoir accès à un mode de réglage avancé des paramètres, s'il le désire. Ces derniers seront fixés à des valeurs par défaut pour les débutants.

Résultats

Enfin, la visualisation des résultats devra apparaître de façon claire et conviviale à l'utilisateur au travers de l'interface Web. De plus, les résultats pourront être exportés dans un fichier pour une analyse ultérieure. Par ailleurs, nous allons essayer de mettre en place un dispositif d'annotations des fichiers.

Aide en ligne

Un message d'aide apparaîtra au survol du curseur de la souris sur la fonction ou l'option choisie. Ainsi l'utilisateur pourra avoir plus d'informations sur la commande concernée.

Le logiciel *regEfmtool* étant en anglais, nous avons choisi d'utiliser cette même langue pour notre interface. De plus, cela permettra son utilisation par un plus grand nombre d'utilisateurs.
Lors de l'affichage de la page d'accueil, l'utilisateur aura le choix entre créer un réseau métabolique manuellement, ou bien le charger à partir de fichiers préexistant.

Création d'un nouveau réseau

Si l'utilisateur clique sur le bouton de création d'un nouveau réseau métabolique, une première page Web s'affichera. Il devra rentrer les noms de tous les métabolites et enzymes participant aux réactions, ainsi que celui du réseau. Il aura la possibilité de valider ou supprimer chaque métabolite et enzyme. Dans le cas d'une validation, les données seront récupérées afin de générer automatiquement les fichiers nécessaires au fonctionnement du logiciel.

Une fois les noms des composants du réseau métabolique entrés, l'utilisateur va devoir créer les réactions une à une. Tout d'abord, il devra s'occuper des réactifs de la réaction : un menu déroulant lui donnera le choix entre tous les métabolites qu'il a saisi précédemment et il n'aura plus qu'à entrer le coefficient stœchiométrique associé. Il en sera de même pour les enzymes (sans le coefficient de stœchiométrie dans ce cas), les produits et les co-facteurs. Pour ce qui est de la réversibilité des

réactions, il suffira de cocher la bonne case. L'utilisateur pourra ensuite ajouter une réaction ou valider son réseau.

Chargement d'un réseau préexistant

Si l'utilisateur clique sur le bouton de chargement d'un réseau depuis la page d'accueil, il devra charger une série de fichiers (depuis le disque dur de son ordinateur) nécessaires au bon fonctionnement de *regEfmtool*. Les fichiers doivent être dans le format adéquat et un message d'erreur apparaîtra si ce n'est pas le cas. Au cas où, une fonction d'aide sera disponible pour avoir un modèle de fichiers à charger.

Lancement du programme

Lorsque l'utilisateur aura créé son réseau manuellement ou l'aura chargé, il devra ensuite choisir les paramètres de calcul de *regEfmtool*. Nous avons fait le choix d'utiliser des cases à cocher en fonction de ce qu'il choisira. Pour les utilisateurs non expérimentés, les choix de base seront pré-sélectionnés. Il suffira ensuite de cliquer sur le bouton "Run" pour avoir les résultats générés par le logiciel. Nous n'avons pas représenté toutes les options du lancement de *regEfmtool* sur cette maquette, pour des raisons d'affichage.

Affichage des résultats

Il sera composé d'une série de "boîtes" dans lesquelles seront affichés les différents éléments générés par *regEfmtool*. Il sera également possible de lancer une comparaison des résultats entre deux réseaux en appuyant sur le bouton "Compare".

Comparaison des résultats

Pour la comparaison des réseaux, il y aura un affichage en vis-à-vis des résultats.

1.3.2 Besoins non fonctionnels

Portabilité

L'utilisation de *regEfmtool* s'appuie sur d'autres logiciels, nécessitant par exemple la version 1.7 de Java. De ce fait, ils devront être libre d'utilisation pour le secteur académique. L'interface devra être livrée avec tous les fichiers de configuration (pré-existants ou nouvellement créés) et indépendante du système d'exploitation.

Sécurité et robustesse

Il faudra gérer l'espace occupé par les fichier chargés sur le serveur. Le site devra être stable et gérer au mieux les erreurs qui pourraient être générées lors du chargement des fichiers, de la modification des paramètres ou des calculs. L'utilisateur sera donc informé en cas d'erreur lors du chargement d'un fichier non compatible avec *regEfmtool*, contenant des erreurs d'écritures ou lorsque les paramètres entrés ne sont pas en accords avec la fonction ou l'option sélectionnée.

Temps de calcul

Il faudra effectuer une vérification du nombre de métabolites et de réactions afin d'estimer le temps de calcul. Si ce dernier s'avère trop long, l'utilisateur sera prévenu et devra confirmer le lancement du processus.

Documentation

L'écriture du code sera constituée de commentaires qui permettront la maintenance du code ainsi qu'une éventuelle amélioration de ce dernier par un tiers. L'interface Web créée, quand à elle, devra être fournie avec une documentation sur son installation, son utilisation, sa maintenance et une charte graphique déclarant les différents attributs du site (couleurs utilisées, police, logo, image,...).

Chapitre 2

Conception

2.1 Règles générales

Lorsque l'utilisateur a choisi de créer un nouveau réseau métabolique, il doit créer les règles qui permettront de le définir. Cette étape n'est possible que lorsque l'utilisateur a créer son réseau et donné les réactions réversibles.

Tout d'abord, l'utilisateur choisi le nombre de réactions qui composent la règle qu'il veut créer. Une fois ce choix effectué, il clique sur le bouton "ok". S'affichent alors deux menus déroulant par réaction. Le premier permet de choisir la réaction et le second sa valeur (0 ou 1).

L'utilisateur ne peut sélectionner le nom de la deuxième réaction que lorsqu'il a choisi la première, ce qui l'empêche de choisir deux fois la même réaction. Le même principe s'applique a chaque nouvelle réaction choisie.

Enfin, quand l'utilisateur a sélectionné toutes les réactions ainsi que leur valeur, il clique sur le bouton "Ajouter", la règle est alors écrite dans le fichier *grfile*.

Chapitre 3

Réalisation

3.1 Règles générales

La page permettant la saisie des règles générales est obtenue à l'aide du fichier *generules.php*.
L'affichage à l'ouverture de la page n'est composé que d'une zone de texte et d'un bouton "ok" qui est de type submit. Au clic, il fait appel à la fonction *add_ reaction()* qui créée deux menus déroulant par réaction jusqu'à atteindre le nombre de réactions entrées par l'utilisateur. Elle vérifie également que le nombre de réactions entrées par l'utilisateur est au moins égal à 2, sinon elle affiche un message sous la forme d'une alert à l'utilisateur.
Les réactions sont récupérées à partir de ? ? ?.

Lorsque l'utilisateur a choisi toutes ses réactions et leur valeur, il clique sur le bouton "Ajouter". Celui-ci fait appel à la fonction *validateForm()* qui vérifie que tous les champs ont bien été sélectionnés. Si la fonction retourne VRAI, il fait appel au fichier *createGrfile.php* qui écrit la règle dans le fichier grfile.
Le fichier *createGrfile.php* écrit dans le fichier *grfile* selon certaines règles. En effet, l'écriture de la règle dépend des valeurs associées aux réactions.

Si la valeur de la réaction qui suit le "THEN" est 0, alors il y aura le symbole "!" au début de la règle (juste après le "="), sauf si la règle ne contient que deux réactions.
Si les réactions après "IF" ou "AND" ont pour valeur :

- 0, alors on écrira (!0reac)
- 1, alors on écrira (!1reac)

En revanche, si la valeur de la réaction après "THEN" est 1, les réactions après "IF" et AND" s'écriront :

- 0reac si sa valeur est 0
- 1reac si sa valeur est 1

De plus, le "AND" affiché sur la page web sera écrit &. Quand la règle est composée d'au moins trois réactions, des parenthèses sont ajoutées après chaque réaction sauf la première et la dernière. Des parenthèses entourent l'ensemble des réactions situées après le "=".

Pour mieux comprendre l'écriture du fichier *grfile*, voici un exemple :
Choix de l'utilisateur sur la page web :
IF R1 valeur : 1
AND R2 valeur : 0
AND R3 valeur : 0
THEN R4 valeur : 0

Règle écrite dans le fichier :
R4 = (!((1R1 & (!0R2)) & (!0R3))

Bibliographie

[1] http://bioinformatics.oxfordjournals.org/content/20/2/226.

[2] http://www.csb.ethz.ch/tools/efmtool.

[3] Christian Jungreuthmayer, David E. Ruckerbauer, and Jürgen Zanghellini. Utilizing gene regulatory information to speed up the calculation of elementary flux modes. 2012.

[4] Axel von Kamp and Stefan Schuster. Metatool 5.0 : fast and flexible elementary modes analysis. *Bioinformatics*, August 2006.

[5] http://www.mpi-magdeburg.mpg.de/projects/cna/cna.html.

[6] Roland Schwarz, Patrick Musch, Axel von Kamp, Bernd Engels, Heiner Schirmer, Stefan Schuster, and Thomas Dandekar. Yana – a software tool for analyzing flux modes, gene-expression and enzyme activities. *BMC Bioinformatics*, June 2006.

[7] http://yana.bioapps.biozentrum.uni-wuerzburg.de/.

[8] S. Pérès, F. Vallée, M. Beurton-Aimar, and J.P. Mazat. Acom : A classification method for elementaryfluxmodes based on motif finding. *Biosystems*, March 2011.