DESS Compétence Complémentaire en Informatique
Faculté des Sciences et Techniques
Parc de Grammont
37000 - TOURS

Développement d'un outil de visualisation

de voies métaboliques

Rapport de stage de DESS
Agathe Madeleine

Responsable : Mme Anne Morgat
Chercheur associé à l'INRIA Rhône-Alpes

Institut National de Recherche en Informatique et en Automatique
INRIA Rhône-Alpes
Projet Hélix

Année 2002

Remerciements

Je remercie vivement Monsieur François Rechenmann, directeur de recherche et responsable du projet Hélix de l'INRIA Rhône-Alpes, pour m'avoir reçue dans son équipe et m'avoir permis de travailler dans un aussi beau cadre avec vue sur les Alpes.

Je remercie Monsieur Alain Viari, directeur de recherche au sein de l'équipe Hélix, pour ses conseils en mécanique qui m'ont été très utiles dans la modélisation des systèmes physiques de mon algorithme.

Je remercie Madame Anne Morgat, chercheur associé au sein de l'équipe Hélix, pour m'avoir prise sous son aile et m'avoir permis de faire mes premiers pas dans un projet bioinformatique.

Je suis très reconnaissante à Stéphane Bruley pour m'avoir encadrée dans mon travail tout au long de mon stage et je le remercie pour ses conseils de développeur avisé et son professionnalisme. Il m'a transmis de vraies méthodes de travail qui m'ont permis de transformer mes connaissances en compétences.

Et enfin, merci à tous les membres du projet Hélix, pour leur accueil, leur sympathie, l'aide qu'ils m'ont apportée tout au long de mon stage et la très bonne ambiance de travail qui règne dans l'équipe. Je vous souhaite à tous une très bonne continuation !

Table des matières

I. Introduction 4
II. Contexte et objectif 5
A. Le métabolisme et les voies métaboliques (pathways) 5
B. Représentation graphique de voies métaboliques 9
1. Cahier des charges 9
2. Etat de l'art 9
III. Représentation d'une voie métabolique 10
A. Abstraction d'une voie métabolique à un graphe 10
B. Choix d'implémentation 12
IV. Affichage automatique d'un graphe 13
A. Présentation d'algorithmes de placement automatique d'un graphe 13
1. Méthode « ressort et charges électriques » 13
2. Méthode du barycentre 14
B. Modèle physique de la méthode des ressort et des charges électrique 14
1. Le système masses-ressort 14
2. Le système d'interaction électromagnétique 16
3. Calcul des positions des particules. 17
C. Description de l'algorithme 17
1. Fonctionnement général de l'algorithme 18
2. Le système d'interaction électromagnétique. 20
3. Le système masses-ressort 21
4. Extension de l'algorithme pour la prise en compte d'un motif 21
5. Translation des nœuds 25
V. Application : PathwayBean 26
A. Implémentation 26
1. Les outils de développement 26
2. Architecture de PathwayBean 26
B. Implémentation de l'algorithme de placement 30
C. Exemple de représentation d'une voie métabolique avec PathwayBean 30

I. Introduction

Le thème fédérateur des travaux de l'équipe HELIX (Informatique et Génomique) est la modélisation des connaissances biologiques liées à la génomique. Le principe est de chercher à représenter explicitement et formellement toute entité biologique qui est manipulée par les programmes d'analyse et de visualisation. Cette représentation est un préalable indispensable à l'exploitation des données issues de la génomique.

Le projet **Panoramix** développé au sein de l'équipe HELIX a pour objectif la conception et l'implémentation de bases de connaissance dédiées à l'exploitation de génomes bactériens entièrement séquencés. Les activités de modélisations sont centrées autour des trois problématiques suivantes :

- la modélisation des données de génomique et de génomique comparative
- la modélisation du métabolisme intermédiaire
- la modélisation du protéome.

D'un point de vue pratique, le développement de ces thèmes s'est concrétisé par la conception et l'implémentation de trois bases de connaissances : Genomix, Metabolix et Proteix. Ces bases ont été développées à l'aide d'un système orienté objet. La figure présentée en annexe décrit schématiquement la complémentarité de ces bases ainsi qu'un exemple d'application.

L'exploitation des données de ces différentes bases par des utilisateurs biologistes nécessite le développement d'un environnement utilisateur. Une fonctionnalité minimale de cet environnement est de permettre la consultation des données et leur représentation sous forme d'interfaces graphiques spécialisées.

Ce stage a pour objectif le développement d'une interface de visualisation permettant de représenter des données du métabolisme cellulaire et en particulier les voies métaboliques de la base Metabolix. Cette interface doit s'intégrer dans l'ensemble des développements logiciels réalisés au sein de l'équipe Helix. Ce doit être un composant graphique Java (développé sous forme de module JavaBean) qui doit pouvoir être utilisé ultérieurement par d'autres développeurs et par des biologistes (utilisateur finaux).

II. Contexte et objectif

A. Le métabolisme et les voies métaboliques (pathways)

Le **métabolisme** est le processus global qui assure aux organismes vivants l'apport et l'utilisation de l'énergie libre dont ils ont besoin pour assurer leurs fonctions variées (biosynthèse de molécules complexes, accomplissement de travail mécanique, transport actif de molécules contre des gradients de concentration,…).

D'un point de vue chimique, le métabolisme peut être décrit par un ensemble de réactions chimiques. Une **réaction chimique** (Figure 1) est la transformation d'un ou plusieurs composés (**substrat(s)**) en un ou plusieurs composés (**produit(s)**). On appelle **métabolites** tous les composés intervenant comme substrat ou comme produit dans une réaction. Généralement, le nombre de substrats (ou de produits) différents intervenant dans une réaction varie de un à trois. Une réaction **irréversible** est une réaction qui ne se fait que dans un sens, alors qu'une réaction réversible est une réaction pouvant se faire dans un sens ou dans l'autre (Figure 1).

La particularité des réactions chimiques qui se déroulent à l'intérieur d'une cellule est d'être catalysée (accélérée) par un type particulier de protéines, les **enzymes**. On appellera « **réaction biochimique** » une transformation chimique catalysée par une enzyme (Figure 1).

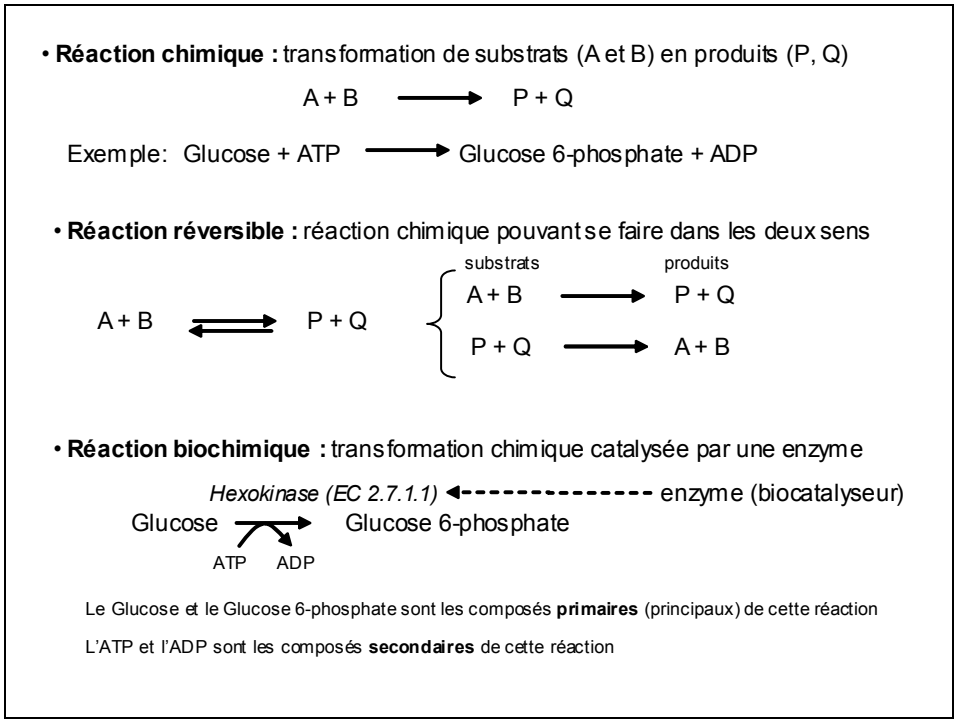


Figure 1 : définition et exemples de réactions chimiques rencontrées dans le métabolisme

Bien que soumises aux mêmes lois de la nature que les autres substances, les enzymes diffèrent des catalyseurs chimiques classiques sur plusieurs points importants (vitesse, conditions de réactions, spécificité vis-à-vis des métabolites, régulation).

Les enzymes catalysent des réactions chimiques très variées. Certains types de réactions nécessitent cependant la présence de molécules supplémentaires appelés cofacteurs (ou encore coenzymes). Un **cofacteur** est une petite molécule, associée transitoirement ou non à l'enzyme, qui participe à la réaction chimique. Un cofacteur peut intervenir comme catalyseur ou comme co-substrat ou co-produit de la réaction. On les considère alors comme des **composés secondaires** de la réaction. Généralement, la transformation de composés secondaires est indiquée par une flèche courbe (Figure 1).

Il existe une nomenclature des enzymes développée par un comité international (*IUBMB : International Union of Biochemistry and Molecular Biology*). Les enzymes sont regroupées par classe en fonction de leur activité catalytique (réaction biochimique catalysée par l'enzyme). A chaque enzyme répertoriée est associé un identifiant (numéro EC).

L'ensemble des réactions biochimiques constituant le métabolisme forme un réseau qui semble d'une complexité désespérante… (Figure 3).

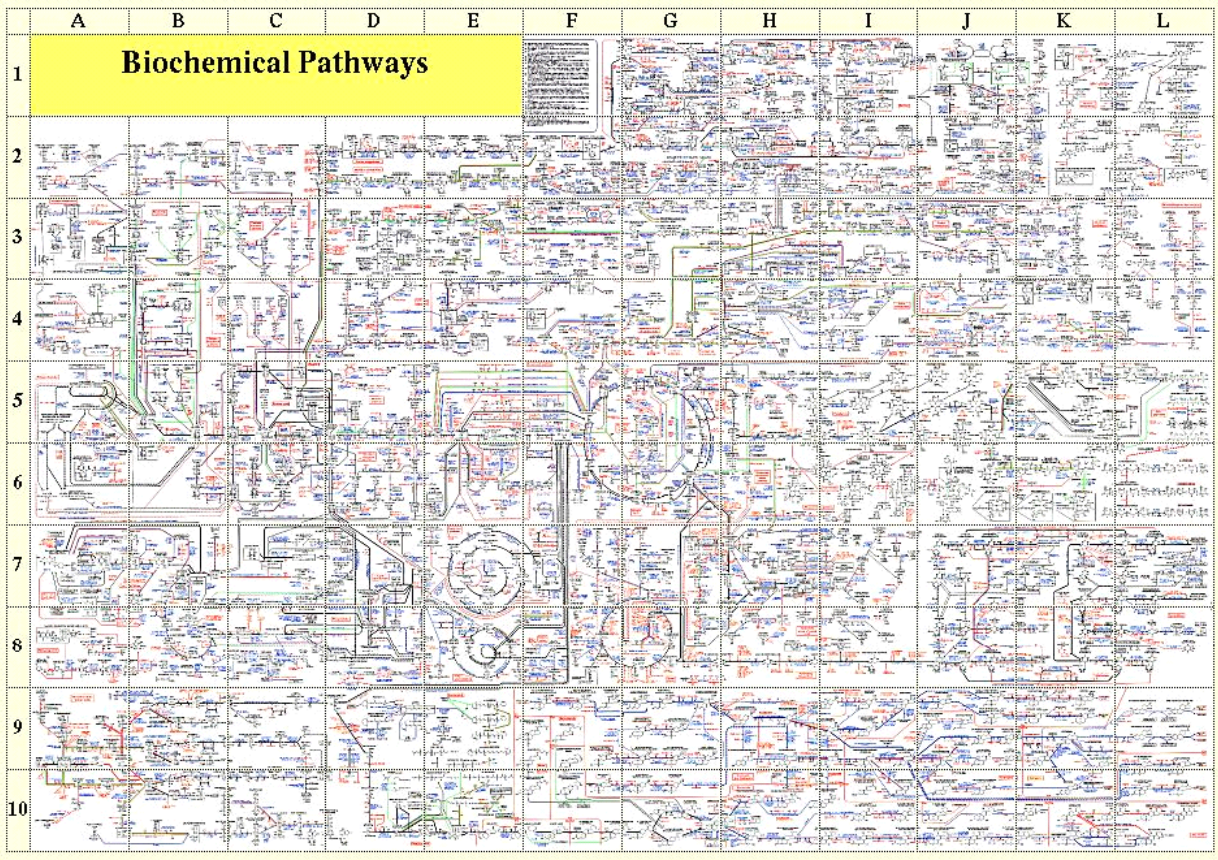


Figure 2 : Carte du métabolisme Boehriger Mannheim [Michal, 1993]
(extrait du site EXPASY : www.expasy.org)

Les biochimistes ont donc l'habitude de travailler sur des domaines plus restreints : les voies métaboliques. Une **voie métabolique**, encore appelée communément ***pathway***, est un ensemble de réactions biochimiques couplées, c'est-à-dire un ensemble de réactions tel que le produit d'une réaction est obligatoirement

substrat d'une autre réaction (excepté pour les substrats et produits initiaux). Il est important de noter qu'il n'existe aucune autre règle permettant de définir plus précisément une voie métabolique. Certains chercheurs distinguent deux grandes catégories de voies dans le métabolisme : celles qui assurent la dégradation (catabolisme) et celles qui assurent la biosynthèse (anabolisme). Par exemple, la glycolyse est la voie de dégradation du glucose. La Figure 3 illustre deux types de représentation graphique de cette voie métabolique. Dans le schéma A, les composés sont représentés par leur nom, alors que dans le schéma B les composés sont représentés par leur nom et par leur formule chimique développée (formule plane).

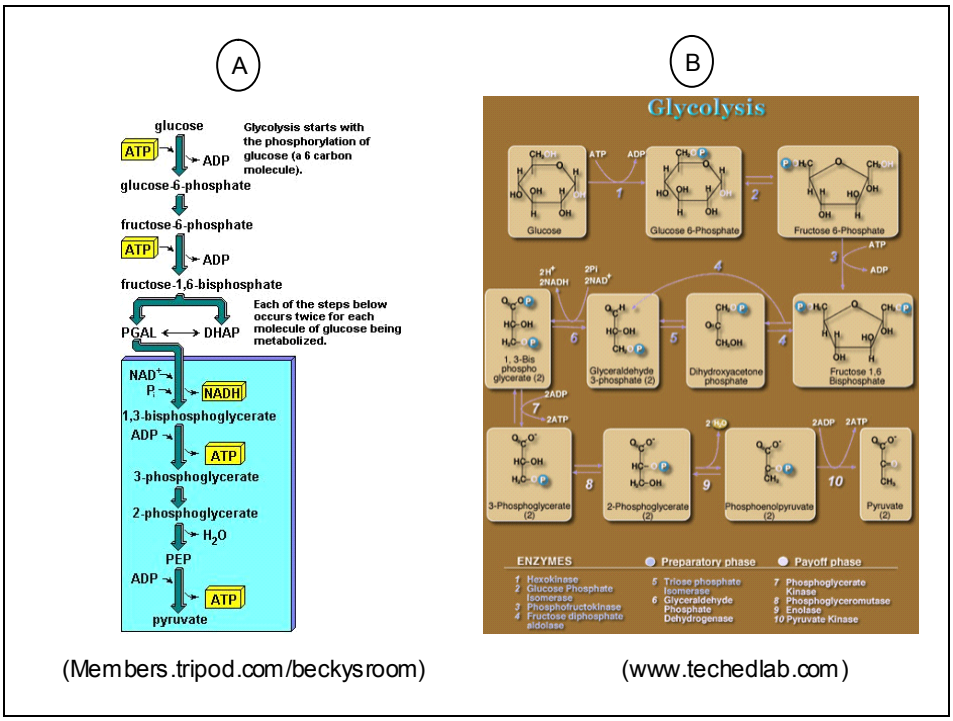


Figure 3 : Deux exemples de représentation graphique de la glycolyse (voie métabolique de dégradation du glucose).

Enfin une voie métabolique peut être linéaire, ramifiée et/ou composée d'un ou plusieurs cycles. Lorsqu'une voie métabolique comporte un cycle, ce dernier est souvent représenté par un cercle ce qui facilite grandement l'interprétation du graphique. A titre d'exemple, les schémas A et B de la Figure 4 représentent deux voies métaboliques qui comportent plusieurs cycles. Leur identification est immédiate pour la voie métabolique représentée par le schéma B alors qu'elle est beaucoup plus difficile pour la représentation graphique adoptée pour le schéma A..

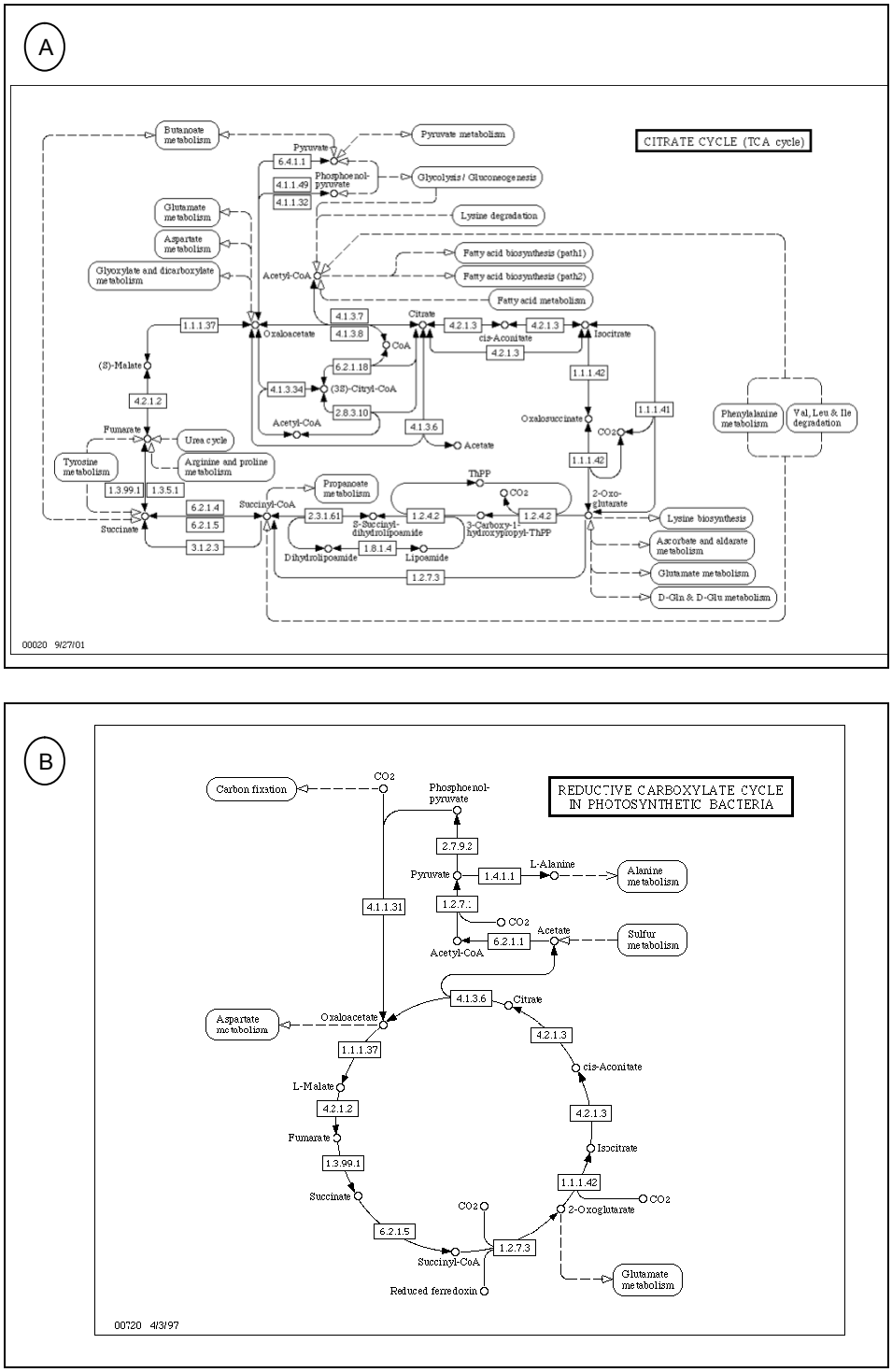


Figure 4: Les deux voies métaboliques représentées par les schémas A et B comportent plusieurs cycles. Leur identification est évidente sur le schéma B mais est beaucoup plus difficile sur le schéma A. Ces deux représentations sont extraites de la base KEGG (**www.genome.ad.jp/kegg**)

B. Représentation graphique de voies métaboliques

Les exemples de voies métaboliques présentées précédemment illustrent les différentes façon de représenter une même information : les métabolites peuvent être représentés par leur formule développée, leur nom usuel, une abbréviation,… Certains auteurs distinguent les composés primaires et les composés secondaires d'une réaction, d'autres ne les distinguent pas ou ne représentent que les composés primaires. Enfin, les enzymes sont éventuellement représentées, soit par leur nom, soit par leur identifiant (numéro EC).

Généralement un diagramme représentant une voie métabolique est dessiné manuellement et sauvegardé sous forme d'image, plus ou moins interactive. C'est le cas de la base KEGG (Kyoto Encyclopedia of Genes and Genomes, www.genome.ad.jp/kegg) [Ogata, 1999] où chaque pathway est associé à une image cliquable, non éditable. L'utilisateur peut accéder ponctuellement à l'information associée à un composé ou à une enzyme.

1. Cahier des charges

Développer un composant graphique Java qui permette de représenter une voie métabolique et de l'afficher. La topologie du diagramme ne doit pas être définie au préalable, mais doit être générée automatiquement. Les spécifications de la représentation graphique sont les suivantes :

- Les composés doivent être suffisamment éloignés les uns des autres pour éviter les chevauchements.
- Les distances entre chaque composé doivent être voisines.
- S'il existe un cycle dans la voie métabolique, les composés participant à ce cycle doivent être représentés sur un cercle. Leurs positions relatives doivent être fixes : le motif cyclique n'est jamais déformé. Cette notion de motif pourrait être étendue à d'autres topologie que les cercles (serpentins, arbres…).

2. Etat de l'art

Quelques projets de bioinformatique ont pour objectif le développement de visualiseurs de pathways métabolique. Le tableau suivant donne la liste des principaux logiciels développés ou en cours de développement.

Logiciels	Description
BioPath biopath.fmi.uni-passau.de/	Logiciel développé par l'université de Passau. Cette application est basée sur Graphlet (www.infosun.fmi.uni-passau.de/Graphlet), un kit de développement d'éditeurs de graphes et d'algorithmes sur les graphes (implémentation C++ et Tcl/Tk). [Schreiber, 2002] [Brandenburg, 2001]
BioCyc ecocyc.org:1555/server.html	Site internet développé par SRI International. BioCyc Knowledge Library (http://biocyc.org/) est une collection de base de données sur les relations pathways/génomes. Les bases BioCyc sont accessibles grâce à un ensemble d'utilitaires de visualisation:"Pathway Tools". Pathway Tools utilise un modèle de données orienté-objet. [Karp, 1994]
PathVis http://www.eml.villa-bosch.de/english/Research/sdbv/1	Logiciel développé par European Media Laboratory (EML). PathVis est un prototype pour la visualisation de voies métaboliques du projet Data Alive (http://www.eml.villa-bosch.de/english/Research/sdbv/1). Ce logiciel, développé en Java, permet une connexion à une base de données. [Rojas, 2002]
PathDB www.ncgr.org/pathdb	Logiciel développé par le NCGR (National Center for Genome Research). Ce projet a pour objectif le développement d'un ensemble de bases de données intégrées (système ISYS) ainsi que des outils de requêtes (QueryTool) et de visualisation (PathwayViewer). Une version prototype est disponible pour les laboratoires académiques mais commercialisée pour les sociétés commerciales.
PathwayBrowser www.yworks.de.en	PathwayBrowser est un outil de visualisation de pathways métaboliques, développé en Java. PathwayBrowser fait partie du package yWays commercialisé par la société yWorks.

Aucun de ces logiciels n'est disponible librement ou développé en Java. Or l'ensemble des développements réalisés dans l'équipe HELIX étant réalisé en Java par souci de portabilité, il a donc été nécessaire de concevoir une nouvelle application.

III. Représentation d'une voie métabolique

A. Abstraction d'une voie métabolique à un graphe

D'un point de vue formel, on peut considérer qu'une voie métabolique (pathway) peut être représentée par un graphe orienté, connexe. Rappelons qu'un graphe orienté G = (V, E) est caractérisé par deux ensembles, V, l'ensemble des nœuds (ou sommets) du graphe et E, l'ensemble des arcs (ou arêtes) e = (i,j) ∈ V x V.

Un graphe est dit connexe lorsqu'il existe au moins un chemin permettant de relier n'importe quel couple de nœuds.

Dans un graphe métabolique, il est nécessaire de distinguer cinq grands types de nœuds (Figure 5) :

1. Les « nœuds composés primaires » représentent les métabolites principaux (substrat(s) et produit(s)).

2. Les « nœuds composés secondaires » représentent les métabolites secondaires (co-substrats, co-produits)
3. Les « nœuds réaction » représentent la réaction biochimique. Cette réaction est souvent assimilée à l'enzyme qui catalyse la réaction.
4. Les « nœuds d'addition » permettent de connecter plusieurs composés intervenant comme substrats OU comme produits dans une réaction.
5. Les « nœuds de liaison » permettent de représenter la connexion entre deux voies métaboliques.

Les arcs de ce graphe ne peuvent relier que des nœuds de types différents. Par exemple, un nœud composé peut être relié à un (ou plusieurs) nœud réaction, par contre deux nœuds composés ne peuvent être directement connectés.

La distinction substrat/produit est représentée par l'orientation de l'arc reliant un « nœud composé » à un « nœud réaction ».

Pour les réactions à plusieurs substrats ou plusieurs produits, les nœuds composés sont reliés à un nœud d'addition, c'est ce dernier qui sera relié au nœud réaction. L'orientation des arcs entre les nœuds composés et le nœud de liaison définit la nature substrat ou produit de l'ensemble des nœuds composés.

Les réactions réversibles sont représentées par deux réactions distinctes : une réaction allant du substrat vers le produit et une autre réaction allant du produit vers le substrat (Figure 5).

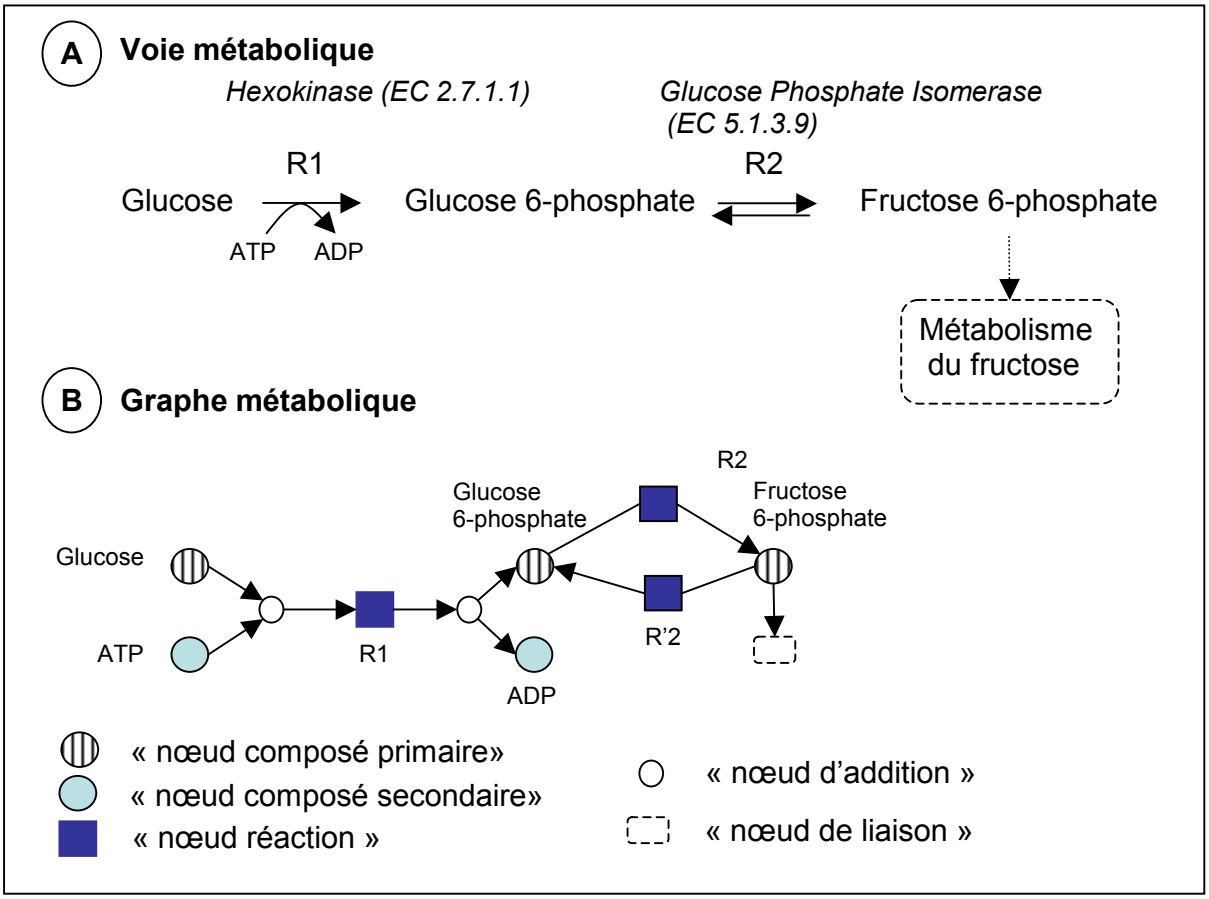


Figure 5 : Un exemple simple de voie métabolique et de son graphe correspondant

Les réactions R2 et R'2 représentent la réaction réversible entre le glucose-6-phosphate et le fructose-6-phosphate.

B. Choix d'implémentation

Il existe des librairies Java d'interface utilisateur qui proposent des composants pour développer des listes, des tableaux, des arbres ainsi que des graphes.
Le tableau suivant représente la liste des librairies évaluées pour la représentation des graphes.

Librairie	Description
OpenJGraph http://openjgraph.sourceforge.net/	OpenJGraph ne présente pas une architecture de classes facilement manipulable pour nos besoins et son développement présente des défauts. De plus, la documentation d'implémentation de la librairie très incomplète.
GEF http://ftp.ics.uci.edu/pub/arch/gef/	GEF (*Graph Editing Framework*) est une autre librairie Java dédiée à la représentation de graphes mais, tout comme OpenJGraph, ce n'est pas l'outil le plus approprié pour l'utilisation que l'on veut en avoir.
JGraph http://jgraph.sourceforge.net/	JGraph est une librairie, distribuée gratuitement sous licence LGPL, permettant de représenter des graphes mathématiques. JGraph est un système extensible: des algorithmes de placement de noeuds du graphe peuvent être facilement intégrés. De plus, la documentation est très complète (documentation d'implémentation et tutoriel d'utilisation).

Nous avons sélectionné la librairie **JGraph** pour le développement de l'application PathwayBean. Cette librairie ne possède aucune méthode de placement automatique des nœuds du graphe, l'étape suivante de ce travail a consisté à rechercher des algorithmes appropriés pour l'affichage automatique d'un graphe métabolique.

IV. Affichage automatique d'un graphe

A. Présentation d'algorithmes de placement automatique d'un graphe

Il existe des algorithmes de placement permettent d'assigner automatiquement des coordonnées aux noeuds d'un graphe.

Les algorithmes de **placement par force** sont les plus appropriés pour représenter un graphe métabolique tel que nous l'avons spécifié au paragraphe précédent. Le graphe est assimilé à un système de particules et de forces interagissant entre ces particules. L'algorithme recherche la configuration de plus faible énergie, c'est-à-dire une position pour chaque particule telle que la somme des forces s'appliquant à cette particule soit la plus faible possible. Décrivons deux méthodes de placement par force.

1. Méthode « ressort et charges électriques »

Cet algorithme est basé sur un système physique (selon la théorie des forces de Hooke) dans lequel les noeuds du graphe sont remplacés par des particules caractérisées par une charge électrique et une masse. Ces particules se repoussent les unes des autres. Les arêtes du graphe sont des ressorts reliant les sommets entre eux. Il s'agit d'une approche heuristique pour dessiner un graphe. La force de tension du ressort appliquée à chaque sommet est proportionnelle à l'allongement du ressort. Entre deux particules, la force électrique appliquée à chaque particule dépend de la charge des particules et de leur distance. Le système évolue d'une configuration initiale à une configuration d'énergie minimale (Figure 6).

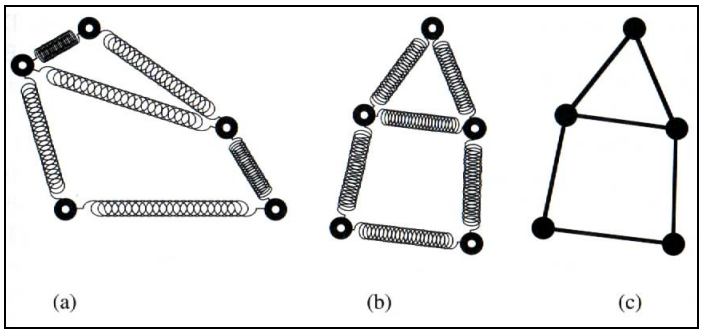


Figure 6 : Evolution d'un graphe dans un algorithme de placement par force

(a) Les arêtes sont remplacées par des ressorts et les sommets par des charges électriques.
(b) Une configuration d'équilibre est atteinte, la somme des forces de chaque particule est nulle.
(c) Graphe représenté sous forme d'un dessin.

Cet algorithme satisfait deux aspects graphiques :

- La force des ressorts permet de positionner les nœuds liés à des distances voisines.
- La force électrique implique que les noeuds ne sont pas trop près les uns des autres.

2. *Méthode du barycentre*

Il s'agit d'une variation de l'algorithme précédent d'après le modèle de Tutte [Tutte, 1963]. Ce modèle utilise des ressorts dont la longueur au repos est nulle. La force appliquée à chaque sommet du graphe est donc directement proportionnelle à la longueur de l'arête liée à ce sommet. La constante de raideur des ressorts n'intervient pas dans le calcul de force et il n'existe pas de forces électriques. Pour des raisons de calcul, il faut diviser l'ensemble des sommets du graphe en deux parties. Le premier sous-ensemble comporte au moins trois sommets fixes. Le deuxième sous-ensemble réunit les sommets libres. Cet algorithme calcule une position, pour chaque sommet libre, située au barycentre de ses voisins. L'un des avantages de cette méthode est que, dans le cas d'un graphe composé de trois connexions, elle aboutit à la représentation d'un polygone convexe.

L'algorithme de placement par force basé sur la méthode des ressorts et des charges électriques paraît le plus approprié pour l'affichage d'un graphe métabolique. Le modèle physique associé à cet algorithme est détaillé dans le paragraphe suivant.

B. Modèle physique de la méthode des ressort et des charges électrique

Les deux modèles physiques (le système de masses-ressort et le système d'interaction électromagnétique) détaillés dans ce paragraphe permettent de calculer les **forces** appliquées à chaque nœud, que nous appellerons **particule**. L'étape suivante consiste à calculer la nouvelle **position** de chaque particule.

1. *Le système masses-ressort*

a) Le ressort

Un ressort a un comportement élastique qui tend à ramener deux particules à une distance donnée l'une de l'autre (Figure 7).

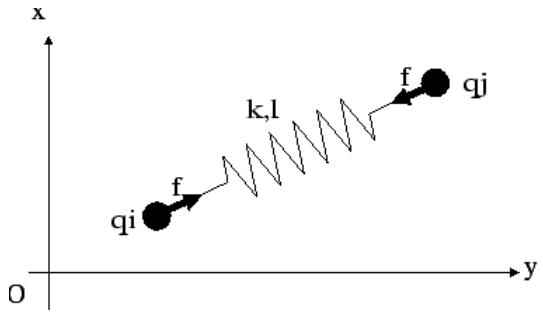


Figure 7: Ressort entre deux particules

Un ressort est caractérisé par une longueur au repos l et une constante de raideur k. La longueur au repos d'un ressort est la longueur pour laquelle le ressort est à l'équilibre.
Les deux particules q_i et q_j exercent des forces réciproques f.

L'action d'un ressort est proportionnelle à son allongement et à sa raideur (Équation 1).

$$f_{j\to i}^{(k)} = k\left(\frac{\|q_i - q_j\| - l}{\|q_i - q_j\|}\right)\frac{q_i - q_j}{\|q_i - q_j\|}$$

Équation 1 : Force de tension d'un ressort

où le premier terme est la constante de raideur, le deuxième terme représente l'allongement relatif (l est la longueur au repos, q_i et q_j sont les positions des particules), et le troisième terme est un vecteur normalisé qui représente la direction de l'effort.

b) L'amortisseur

Un amortisseur sert à réduire le mouvement relatif de deux particules (Figure 8).

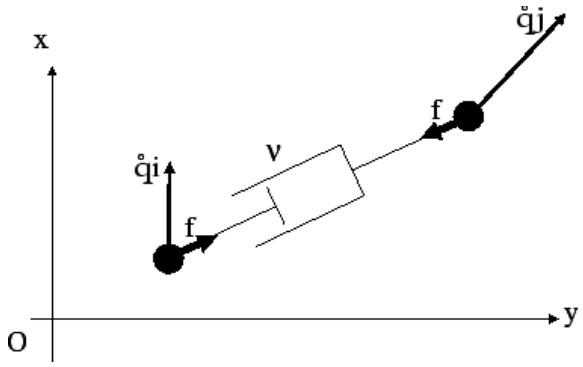


Figure 8 : Amortisseur entre deux particules

Un amortisseur est caractérisé par une constante de viscosité v.
Les vecteurs vitesses des particules qi et qj sont $\dot{q}_i$ et $\dot{q}_j$.

L'action d'un amortisseur est proportionnelle à sa viscosité et à sa vitesse d'allongement (différence des vitesses projetées sur la direction). L'action sur une particule i est donnée par l'Équation 2.

$$f_{j\to i}^{(\nu)} = \nu\left(\frac{\dot{q}_i - \dot{q}_j}{\|q_i - q_j\|}\frac{q_i - q_j}{\|q_i - q_j\|}\right)\frac{q_i - q_j}{\|q_i - q_j\|}$$

Équation 2 : Force d'amortissement

c) Le ressort amorti

Un ressort amorti s'obtient en combinant ressort et amortisseur en parallèle, comme illustré sur la Figure 9.

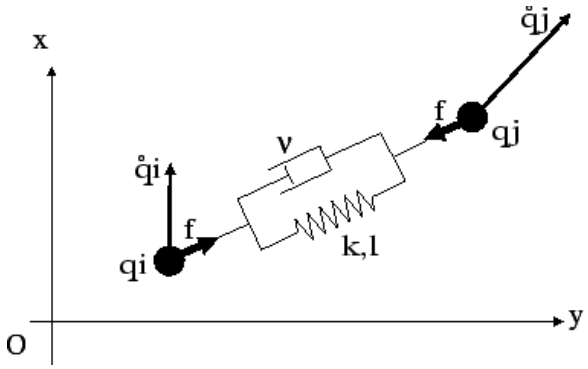


Figure 9 : Ressort amorti

La force appliquée à un ressort amorti est la somme des forces s'appliquant au ressort et à l'amortisseur (Équation 3).

$$f_{j\to i} = f_{j\to i}^{(k)} + f_{j\to i}^{(\nu)}$$

Équation 3 : Force s'appliquant à un ressort amorti

2. Le système d'interaction électromagnétique

L'interaction électromagnétique est la force responsable de l'attraction et de la répulsion entre particules chargées électriquement (Figure 10). La force d'interaction magnétique est décrite par la loi de Coulomb: la force qui s'exerce entre deux particules est inversement proportionnelle à leur distance et proportionnelle au produit des deux charges (Équation 4).

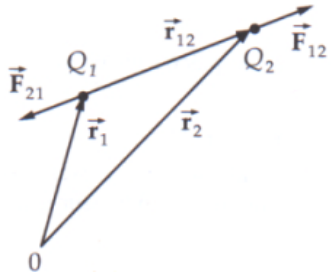


Figure 10 : Schéma d'interaction de deux charges de même signe Q_1 et Q_2.

Q_1et Q_2 sont les charges des particules.
Le vecteur r_{12} est le vecteur direction.
Les vecteurs r_1 et r_2 sont les vecteurs position des charges Q_1 et Q_2.
Les vecteurs F_{12} et F_{21} sont les vecteurs de force d'interaction appliqués respectivement aux charges Q_2 et Q_1.

$$\vec{\mathbf{F}} = \frac{1}{4\pi\varepsilon_0}\frac{Q_1 Q_2}{r^2}\frac{\vec{\mathbf{r}}_{12}}{r_{12}}$$

Équation 4 : Force d'interaction électromagnétique.

Avec :

- r est la distance séparant les deux charges
- ε_0 est la constante diélectrique

3. *Calcul des positions des particules.*

Soit une particule de position q au temps t.

A partir de la force appliquée sur cette particule et de sa masse, on en déduit son accélération.

Soient:

- $\ddot{q}$, l'accélération de la particule
- f, la force s'appliquant à la particule
- m, la masse de la particule

Alors : $\ddot{q} = f / m$

L'accélération de la particule permet de déterminer sa vitesse au temps t+1, à partir de sa vitesse et de son accélération au temps t.

Soient :

- $\dot{q}$, la vitesse de la particule
- h, la longueur du pas de temps

Alors : $\dot{q}(t+1) = \dot{q}(t) + \ddot{q}(t) * h$

Selon le schéma d'intégration d'Euler, sa nouvelle position au temps t+1 est :
$q(t+1) = q(t) + \dot{q}(t) * h$.

C. Description de l'algorithme

Le système masses-ressort et le système de répulsion permettent un affichage du graphe tout d'abord en évitant une trop grande proximité (voire une superposition) de deux noeuds mais aussi en fixant une longueur d'arête minimale entre deux noeuds. Le système tend vers un équilibre en respectant ces deux contraintes.

L'algorithme modélise un système physique à partir d'un graphe représentant l'abstraction d'une voie métabolique. Un nœud du graphe est considéré comme une particule ayant une position, une vitesse, une masse et une charge. Un arc est considéré comme un ressort amorti caractérisé par une raideur, une viscosité et une longueur au repos. Les particules exercent les une sur les autres des forces de répulsion (Figure 11).

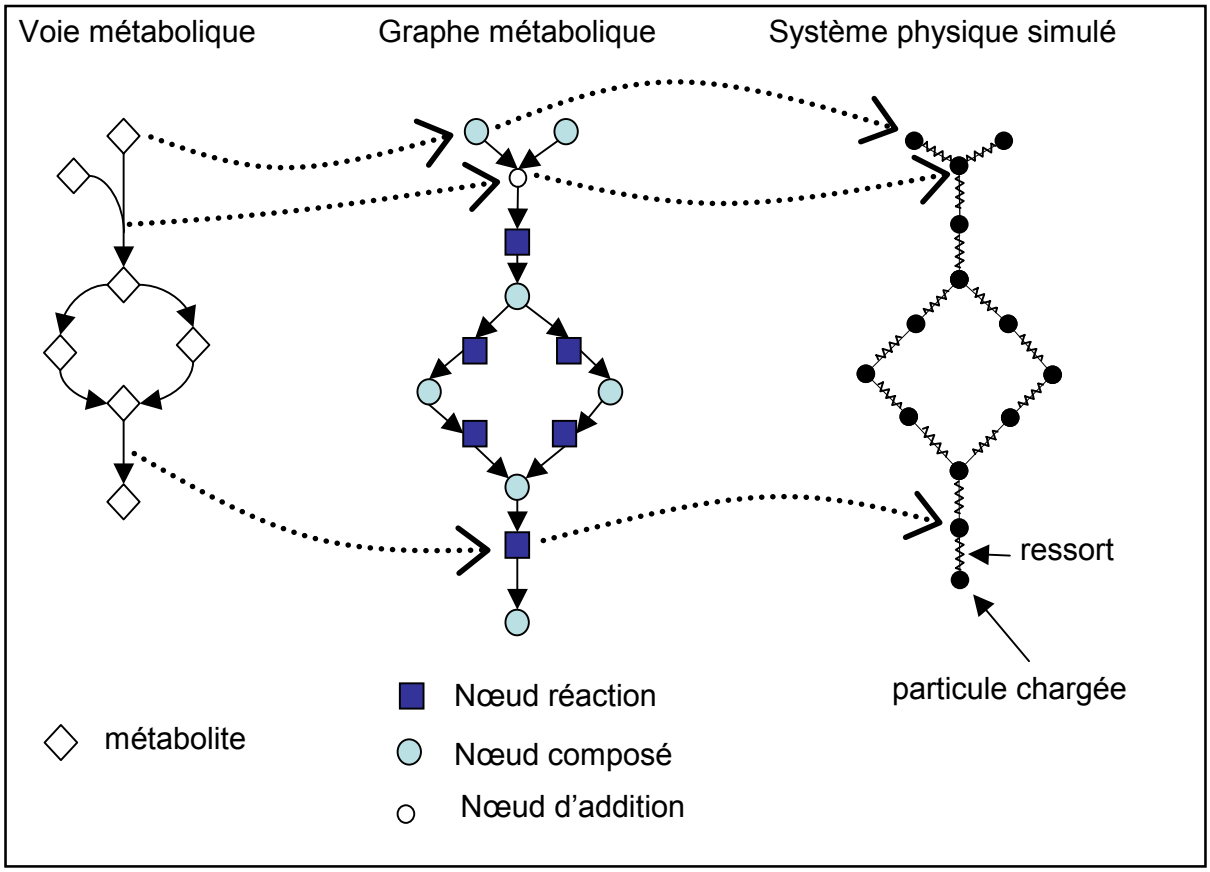


Figure 11 : Une voie métabolique est représentée par un graphe, lui-même modélisé par un système physique. Pour simplifier le schéma, on considère un seul type de nœud composé.

1. Fonctionnement général de l'algorithme

L'algorithme fonctionne sur le schéma suivant : tant que le système physique ne tend pas à l'équilibre, il est nécessaire, pour chaque particule du système, de déterminer sa nouvelle position en tenant compte des interactions avec les autres particules. Aussi, il faut calculer deux forces pour chaque particule.

La première force est la force de tension (et d'amortissement) induite par le ou les ressorts reliés à cette particule (en tenant compte de la particule qui est à l'autre extrémité du ressort : voir le système masses-ressort).

La seconde force à calculer est la force de répulsion induite par l'ensemble des autres particules (voir le système d'interaction électromagnétique). En sommant ces deux forces, on obtient la force résultante à appliquer à la particule. On peut ainsi calculer sa nouvelle vitesse et sa nouvelle position (Figure 12).

Les forces et les vitesses des particules sont stockées sous forme de vecteur. Le repère géométrique utilisé dans les calculs de vecteurs est un repère orthonormé. Pour plus de facilité, nous avons décidé que tous les vecteurs ont pour origine le point (0,0).

```
Procédure LAYOUT
Paramètres : Graphe : g ;
Variables : SystèmePhysique : s,
            Vecteur(taille 2) : force, ft, fi,
            TableauDeVecteurs(taille 2) : V ;
début
    // créerSystèmePhysique crée le système physique correspondant
    // au graphe donné en paramètre
    s ← créerSystèmePhysique(g) ;
    // creation et initialisation du tableau contenant les vitesses
    V ← nouveau TableauDeVecteurs(taille 2)(nombreDeNoeuds(s)) ;
    // boucle principale

    tant que nonEquilibre(s) faire
        pour chaque n_i ∈ noeuds(s) faire
            // initialisation des forces
            force ← (0,0) ;
            fi ← (0,0) ;
            ft ← (0,0) ;

            pour chaque n_j ∈ noeuds(s) tel que n_j relié à n_i par un
            ressort faire
                ft ← ft + calculForceTension(n_i,n_j) ;

            pour chaque n_j ∈ noeuds(s) tel que n_j ≠ n_i faire
                fi ← fi + calculForceRépulsion(n_i,n_j) ;

            // calcul de la force résultante
            force ← ft + fi ;
            // mise à jour de la vitesse du noeud n_i
            V[i] ← V[i] + calculVitesseNoeud(force,n_i) ;

        pour chaque n_i ∈ noeuds(s) faire
            miseAJourCoordonnées(n_i,V[i]) ;

    AppliquerCoordonnéesDuSytèmePhysique(g,s) ;
fin
```

Figure 12 : Algorithme de placement

En général, la force de répulsion étant très faible devant la force de tension du ressort, l'influence de la répulsion sur le comportement de deux noeuds liés par un ressort est négligeable.

Dans le cas particulier de deux noeuds superposés, la distance entre ces deux nœuds est nulle ce qui entraînerait une division par zéro dans le calcul de forces. Afin

d’éviter ce problème, un déplacement vertical d'un des noeuds est effectué immédiatement. La vérification de la superposition, et si nécessaire la translation, ne se fait que dans la partie du système de répulsion. Le système de répulsion doit donc intervenir avant le système masses-ressort.

2. Le système d’interaction électromagnétique.

On ne s'intéresse ici qu'aux effets de répulsion, donc les charges des particules sont toutes de même signe.

On réalise une itération sur l’ensemble des nœuds du graphe. Pour tout couple de nœuds, le premier noeud est appelé source et le second est appelé cible.

Un vecteur représente la distance entre la source et la cible. Le calcul de force d'interaction dépendant de l'inverse de la distance séparant les deux noeuds, il est nécessaire de tester si la distance est nulle.

- Cas où la distance est nulle. On déplace la source selon l'axe des ordonnées. Ainsi, le vecteur de distance possède une norme très faible, ce qui évitera toute division par zéro dans le calcul de force, et éloignera d'autant plus les deux noeuds l'un de l'autre qu'ils sont proches.
- Cas où la distance n'est pas nulle. Afin de limiter la répulsion de deux noeuds éloignés d'une certaine distance, on fixe une longueur maximale d'interaction: le double de la longueur au repos d’un ressort. Ainsi, si la distance séparant les deux noeuds est supérieure à la longueur maximale, aucune force de répulsion n'est appliquée, sinon le calcul de la valeur de cette force est effectué selon la formule de force interaction électromagnétique.

Le calcul de force se décompose en deux parties:

- Calcul de la valeur de la force d'interaction. Il s'agit d'obtenir la norme du vecteur de la force interaction.
- Calcul du vecteur de la force d'interaction. Il s'agit de déterminer la direction et le sens du vecteur de la force interaction. En divisant le vecteur de distance par sa norme, on obtient un vecteur unitaire ayant la même direction et le même sens que le vecteur de distance. Ensuite, il suffit de multiplier la valeur de la force par ce vecteur unitaire pour obtenir le vecteur de force de répulsion.

Le vecteur calculé est le vecteur de la force de répulsion s'appliquant à la cible. Le vecteur de la force de répulsion à appliquer à la source est un vecteur de même direction, de même norme mais de sens opposé à celui de la cible (loi de Newton sur la réciprocité des forces).

A partir des forces s'appliquant sur les deux noeuds, il est possible de déterminer l'accélération des noeuds. En effet, le vecteur de l’accélération d’un nœud est calculé en divisant le vecteur de sa force par sa masse.

Le vecteur de l’accélération permet de déterminer la vitesse, au temps t+1, de chacune des particules à partir de la vitesse et de l'accélération au temps t.

Enfin, les vecteurs des vitesses permettent d'obtenir les nouvelles positions q des noeuds.

3. Le système masses-ressort

Le calcul du vecteur de la force de tension du ressort est composé de trois parties:

- Détermination de la valeur de la force à partir de la formule physique sans en considérer le troisième terme (le vecteur normalisé).
- Détermination de l'angle formé par la direction du vecteur distance et l'axe des abscisses.
- Détermination du signe de la force de tension en fonction de la longueur du ressort. Si cette longueur est supérieure à la longueur du ressort au repos, la force est négative, sinon la force est positive.

Le calcul du vecteur de la force d'amortissement est décomposé en trois parties:

- Récupération de la valeur précédente du vecteur vitesse de chaque noeud (résultant de la répulsion).
- Détermination de la projection du vecteur vitesse de chaque noeud selon la direction du ressort.
- Calcul de la valeur signée de l'amortissement à appliquer sur chaque nœud.

La force totale appliquée au système masses-ressort est la somme des vecteurs force de tension et force d'amortissement.

La force calculée est la force s'appliquant à la cible. Le vecteur force à appliquer à la source est un vecteur opposé au vecteur calculé pour la cible.

4. Extension de l'algorithme pour la prise en compte d'un motif

Un **motif** est un ensemble de noeuds fixes les uns par rapport aux autres. Les noeuds appartenant à un motif sont traités différemment: ils ne subissent aucune force entre eux. Un motif possède un centre calculé en fonction des positions de ses nœuds. Ce centre correspond au barycentre du motif. Par exemple, un cycle dans le graphe peut être considéré comme un motif. Les nœuds qui composent ce motif seront placés sur un cercle. Un noeud faisant partie d'un motif sera nommé **pseudo nœud (PN)**, un noeud n'en faisant pas partie sera nommé **vrai nœud (VN)**. On appelle **centre de gravité (CG)** le centre du motif (Figure 13).

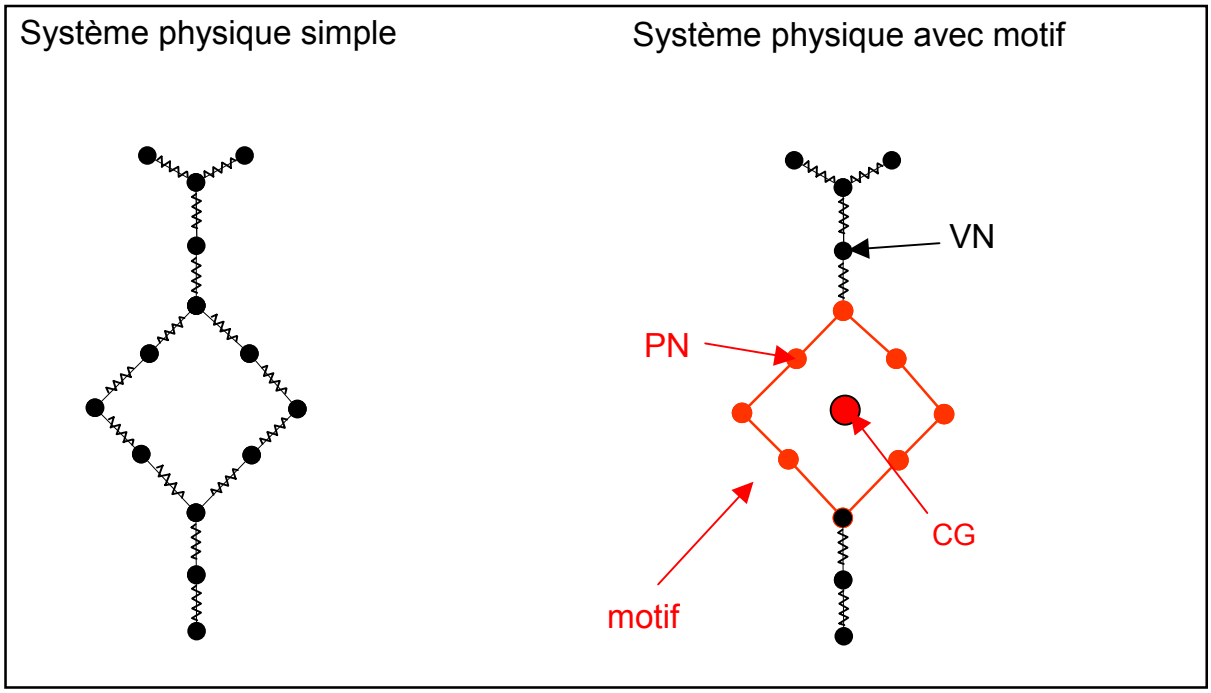


Figure 13 : Système physique comportant un motif

Les nœuds appartenant à un motif sont fixes les uns par aux autres. Il n'existe pas de ressort entre eux.

a) Système d'interaction électromagnétique appliqué au motif

Lors de l'itération sur l'ensemble des nœuds du graphe (cf. paragraphe 2), pour tout couple de nœuds, si la source et la cible font partie du motif, aucun calcul n'est effectué. Sinon, le calcul de la force de répulsion entre un vrai nœud et un pseudo nœud est identique au calcul de force entre deux vrais nœuds (décrit précédemment).

b) Système masses-ressort appliqué au motif

La masse d'un pseudo nœud est celle du motif (soit la somme des masses des noeuds du motif) et sa vitesse est la vitesse du motif.

Un système formé d'un pseudo nœud et d'un vrai nœud est divisé en deux sous-systèmes :

- Un système vrai nœud / mur. Ce système modélise un ressort dont les extrémités sont une particule et un mur, c'est à dire que le système masses-ressort est composé d'un ressort entre une masse légère (le vrai nœud) et un mur (une masse infinie). Les forces physiques appliquées sont la force de tension du ressort entre le vrai noeud et le mur ainsi que la force d'interaction entre ces deux entités.
- un système vrai noeud / centre de gravité. La seule force physique du système est la force de tension du ressort entre le vrai noeud et le centre de gravité.

Prenons l'exemple d'un système composé d'un motif (représentant un cycle) et d'un vrai nœud (Figure 14).

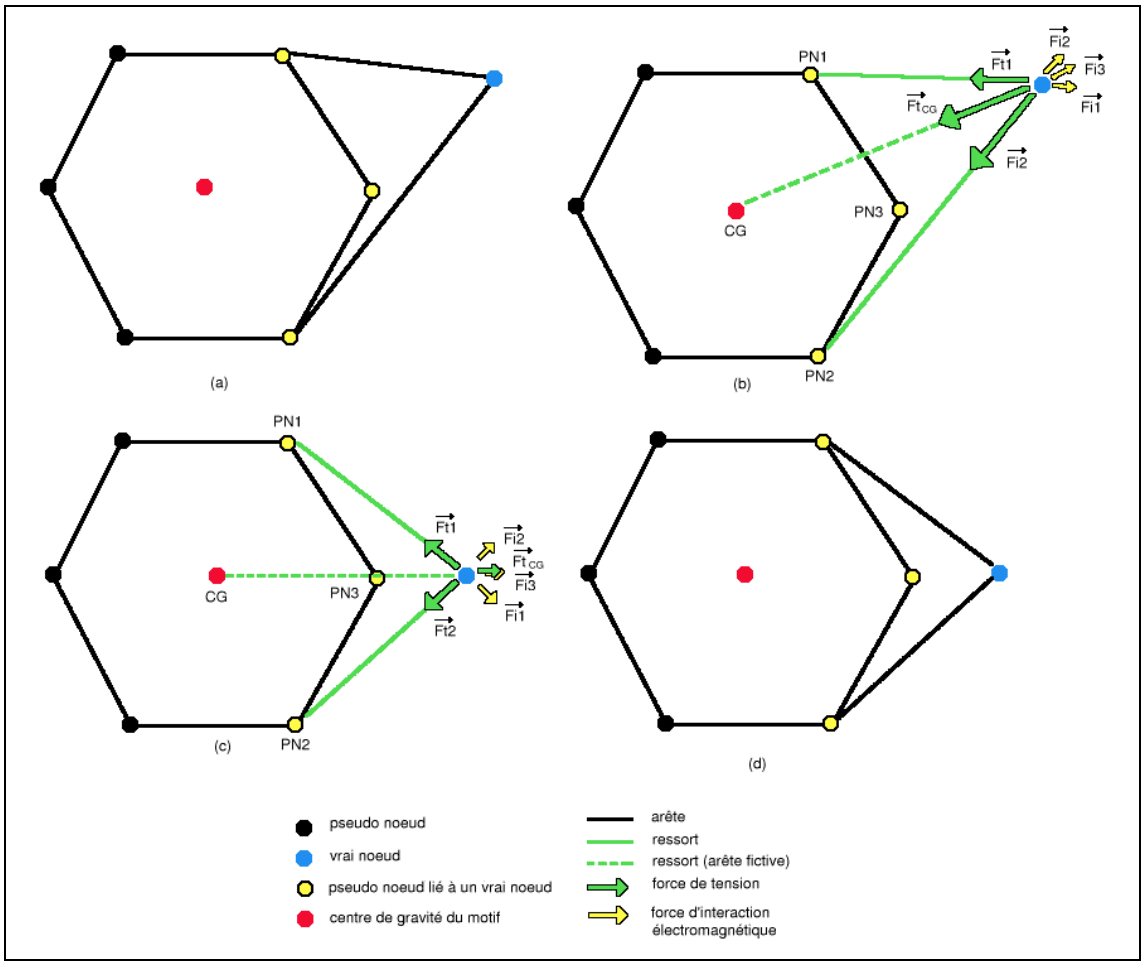


Figure 14 : Evolution d'un système composé d'un motif circulaire et d'un vrai nœud au cours de l'algorithme.

(a) Etat non équilibré du système.
(b) Action des forces de répulsion et des ressorts sur le vrai nœud. Déplacement du vrai nœud.
(c) Etat d'équilibre entre les forces.
(d) Etat stable et final du système.
Par souci de clarté du dessin, les forces de répulsion émises par les trois nœuds placés sur la partie droite du motif ne sont pas représentées.

En combinant les deux sous-systèmes, VN/PN et VN/CG, on obtient une représentation plus appropriée car la position du vrai nœud s'équilibre sur l'axe centre de gravité / pseudo nœud (Figure 15).

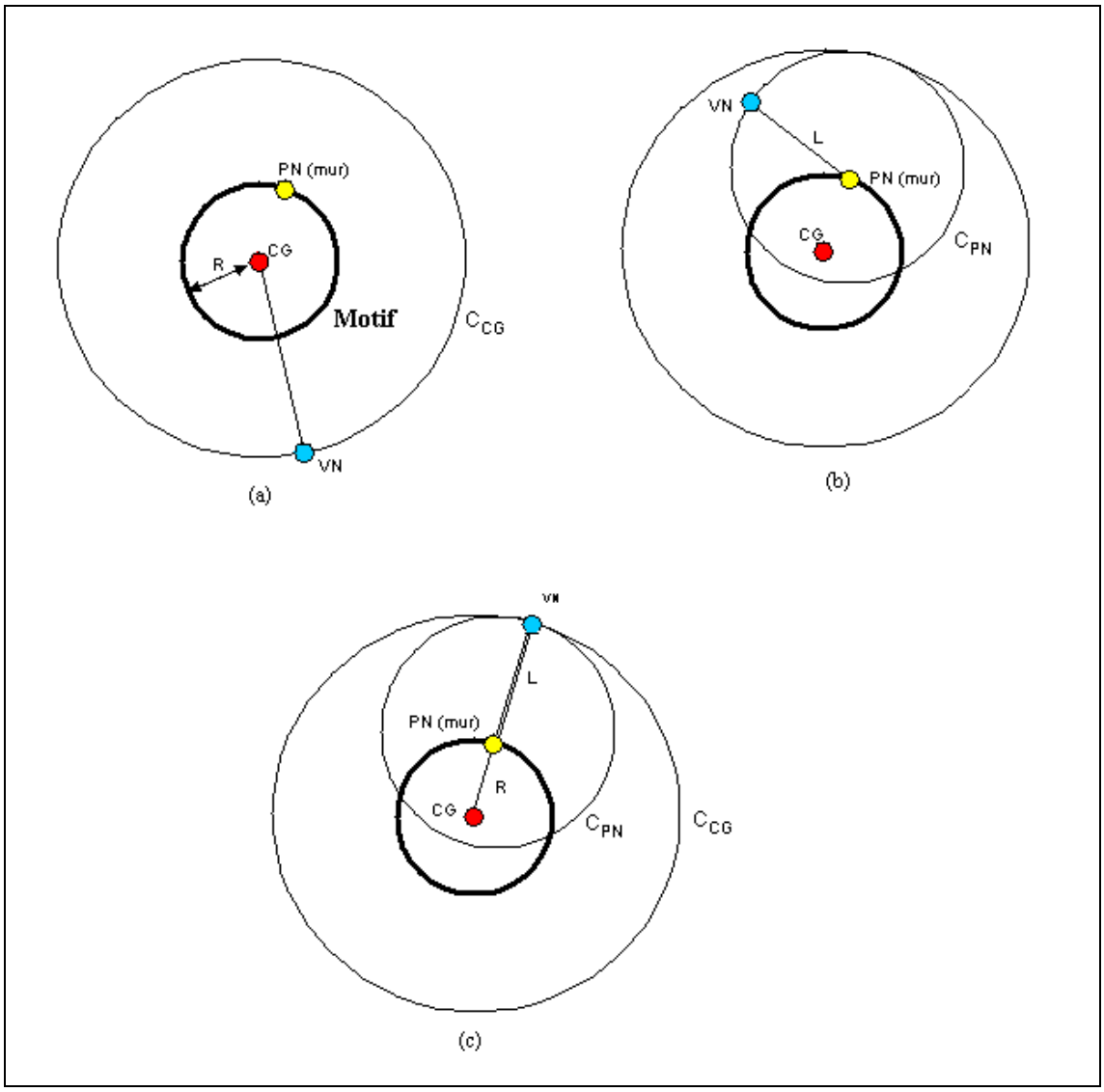


Figure 15 : Le système vrai nœud / pseudo nœud dans un motif circulaire.

(a) Le système vrai nœud / centre de gravité. Le vrai nœud VN peut prendre toutes les positions se trouvant sur le cercle C_{CG} dont le rayon est la longueur du ressort VN – CG (CG est le centre de gravité du motif).

(b) Le système vrai nœud mur. Le vrai nœud peut prendre toutes les positions se trouvant sur le cercle dont le rayon est la longueur du ressort VN – PN (PN est le pseudo nœud). La longueur au repos de ce ressort est L.

(c) Dans un système vrai nœud / pseudo nœud, le vrai nœud est soumis aux forces des deux ressorts VN-CG et VN-PN. Sa position d'équilibre est à l'intersection des deux cercles C_{CG} et C_{PN}. La longueur au repos du ressort VN-CG est R + L.

Dans chaque sous-système (VN/PN et VN/CG), un vecteur de distance est construit de la source vers la cible et le calcul de forces se réalise selon le même schéma que dans un système masses-ressort dépourvu de motif (décrit précédemment). Cependant, les conditions initiales (positions des noeuds, vitesses, masses et longueur de repos) diffèrent d'un sous-système à l'autre.

(1) Le système VN /mur

Le vrai noeud a sa propre vitesse alors que la vitesse du mur est nulle. La masse du vrai noeud et la longueur au repos du ressort sont celles définies par défaut. A l'issue du calcul des vitesses (cf. Schéma général de calcul de forces), la vitesse du vrai noeud n'est pas ajoutée à son ancienne vitesse mais conservée.

(2) Le système VN /CG

La masse du vrai noeud est la masse par défaut. La masse du centre du motif est la somme des masses de chaque noeud composant le motif. La longueur au repos du ressort est égale au rayon du motif ajouté à la longueur au repos par défaut. Ainsi, la position du vrai noeud s'équilibre sur l'axe centre du motif / pseudo noeud. De même que dans le système vrai noeud / mur, la vitesse du vrai noeud n'est pas ajoutée à son ancienne vitesse mais conservée.

5. Translation des nœuds

a) Translation des vrais nœuds

La nouvelle position d'un nœud se calcule à partir de sa vitesse.
Dans l'algorithme, le calcul de la position d'un noeud, au temps t+1, utilise les projections du vecteur vitesse selon l'axe des abscisses et l'axe des ordonnées:

$q_x(t+1) = q_x(t) + \dot{q}_x(t)*h$
$q_y(t+1) = q_y(t) + \dot{q}_y(t)*h$

avec :
q_x, l'abscisse du nœud
q_y, l'ordonnée du nœud
$\dot{q}_x$, la vitesse du nœud selon l'axe des abscisses
$\dot{q}_y$, la vitesse du nœud selon l'axe des ordonnées

b) Translation du motif

La nouvelle position du centre de gravité du motif est calculée à partir de sa vitesse, comme dans le cas d'un vrai noeud. La position de chaque noeud du motif est calculée en ajoutant la position initiale du pseudo noeud au déplacement du centre du motif.

V. Application : PathwayBean

PathwayBean est un composant graphique java permettant le dessin automatique d'un graphe métabolique tel que nous l'avons défini au paragraphe III.A.

A. Implémentation

1. *Les outils de développement*

Les outils de développement utilisés pendant le stage sont les suivants :

- Java, langage orienté-objet et multi-plateforme.
- Ant, outil de production de logiciel Java/XML. Ant est utilisé pour la compilation et l'exécution de programme Java ainsi que pour la génération de la documentation en formats PDF, HTML ou RTF à partir de sources écrites au format XML.
- CVS (Concurrent Version System), outil permettant de gérer les différentes versions d'une application qui est utilisée lorsque plusieurs développeurs travaillent sur un même projet.
- Emacs, éditeur de texte reconnaissant plusieurs formats (éditeur XML, environnement de programmation Java…)

2. *Architecture de PathwayBean*

L'architecture de PathwayBean est basée sur celle de JGraph et suit le concept Modèle, Vue, Contrôleur.

a) Le concept Modèle, Vue, Contrôleur (MVC)

Il s'agit d'un concept employé dans les applications orientées objets. Le modèle MVC permet de créer une séparation entre les données (que l'on veut représenter) et leur représentation.

Le principal avantage du concept Modèle, Vue et Contrôleur est un ensemble de frontières d'encapsulation, qui permet de réduire les interdépendances et améliore la compréhension du système.

Le **modèle** représente les données sur lesquelles on veut travailler.
La **vue** correspond à la représentation graphique des données.
Le **contrôleur** gère l'interaction entre la vue et le modèle. C'est une interface interne réalisant l'association dynamique entre l'objet et sa représentation (Figure 16).

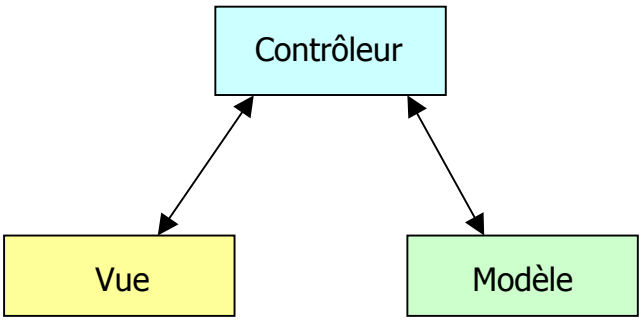


Figure 16 : Interactions entre chaque unité du modèle MVC.

Une vue correspond à un modèle et un modèle correspond à une vue.
Plusieurs vues peuvent correspondre à un modèle.

En général, dans un système MVC, les méthodes associées à un modèle sont des méthodes d'accès aux objets. Les méthodes associées à une vue sont des méthodes d'accès au contrôleur et des méthodes d'affichage. Les méthodes associées à un contrôleur sont des méthodes d'accès à la vue et au modèle.

Le modèle possède quatre types de méthodes :

1. Interrogation de son état interne (lire).
2. Manipulation de son état interne (modifier, détruire).
3. Ajouter ou enlever des écouteurs.
4. Exécuter des événements.

La vue et le contrôleur possèdent trois types de méthodes :

1. Mise à jour de l'apparence.
2. Retourner des informations géométriques.
3. Gérer les événements de l'interface.

b) Architecture des classes

La spécialisation de certaines classes de JGaph permet d'obtenir des classes plus appropriées pour PathwayBean, notamment pour créer les divers types de cellules (composé primaire ou secondaire, réaction, lien vers une autre voie métabolique).

Le modèle du graphe fait référence à la classe PathwayGraphModel. La vue du graphe est utilisée dans un contexte d'objet issu de PathwayGraphView (Figure 17).

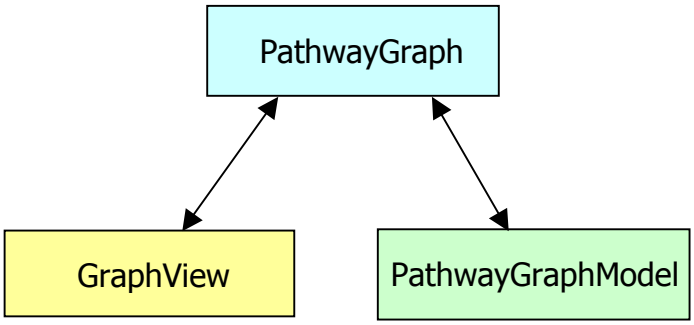


Figure 17 : modèle MVC de PathwayBean.

GraphView est une classe de JGraph.

Le modèle fournit les données du graphe, des informations sur les cellules qui composent ce graphe. Les cellules peuvent être des nœuds (ou sommets) ou des arêtes.

Une première approche a abouti à une organisation des modèles des cellules telle que les classes représentant les composés primaires, PrimReactantNode, et secondaires, SecReactantNode, héritent d'une même classe : ReactantNode (Figure 18).

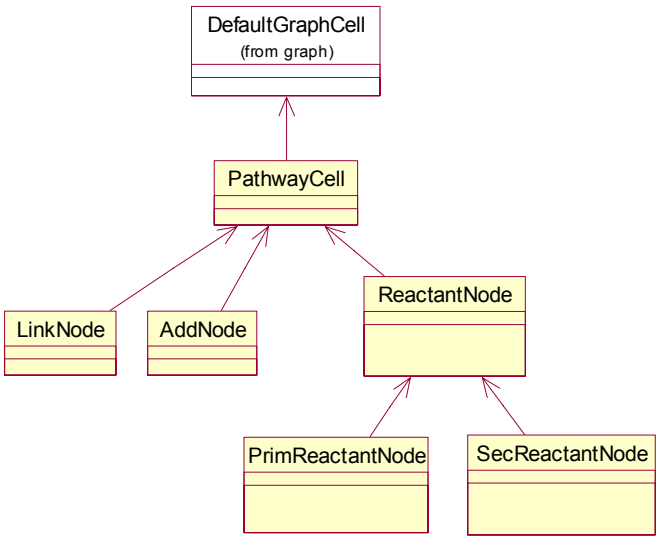


Figure 18 : Première organisation des modèles des cellules (diagramme réalisé avec Rational Rose).

DefaultGraphCell est une classe de JGraph.

Finalement, une seconde architecture a été choisie. En effet, à cause d'une nouvelle spécification des attributs de chaque classe de composé, la classe ReactantNode n'avait plus d'intérêt.

Les différents types des nœuds du graphe sont des instances de PathwayCell et les représentations de ces nœuds sont des instances de PathwayCellView. Une arête est une instance de PathwayEdge et sa représentation est une instance de PathwayEdgeView (Figure 19 , Figure 20)

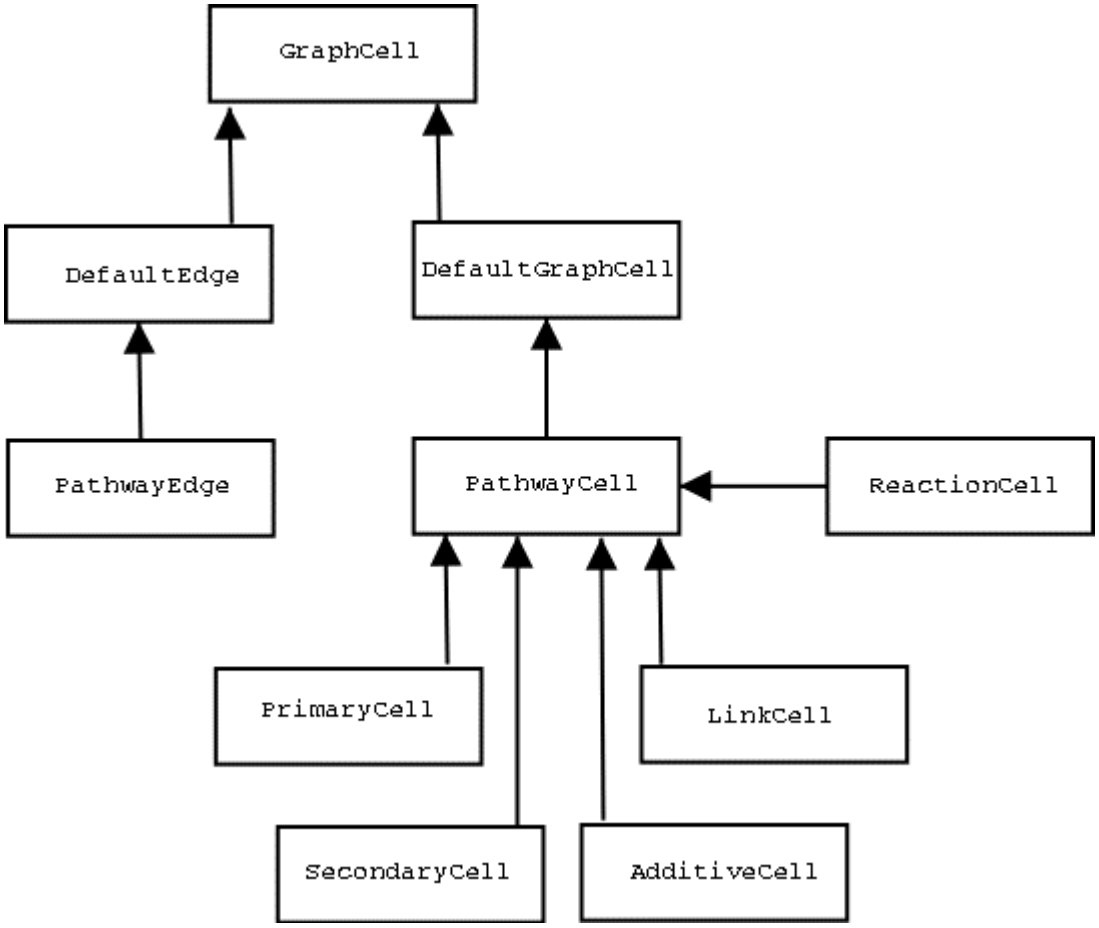


Figure 19 : Modèles des cellules de PathwayBean.

Les arêtes (PathwayEdge), les composés primaires (PrimaryCell), les composés secondaires (SecondaryCell), les réactions (ReactionCell), les cellules de lien vers une autre voie métabolique (LinkCell), les cellules permettant l'ajout d'un composé à une réaction (AdditiveCell) sont issues d'une même classe GraphCell.
GraphCell, DefaultEdge, DefaultGraphCell sont des classes de JGraph.

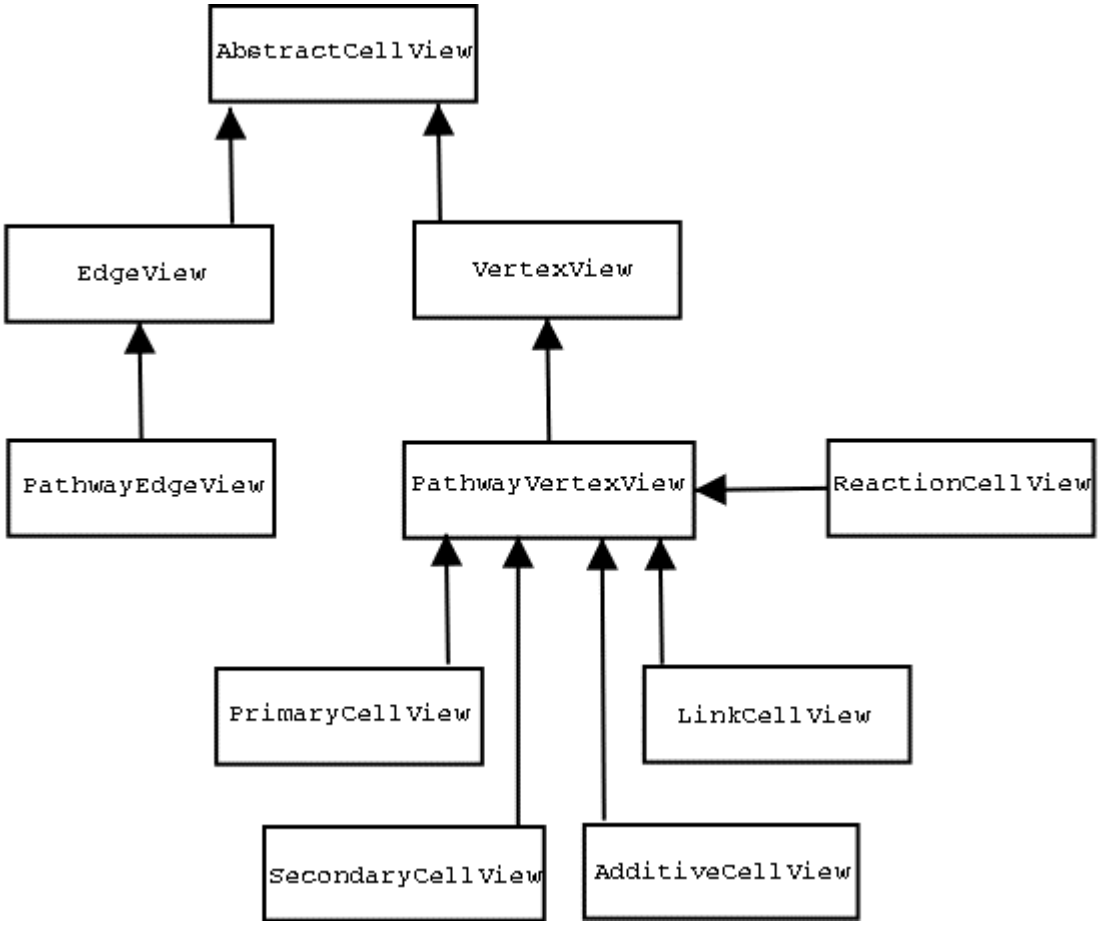


Figure 20 : Vues des cellules de PathwayBean.

Les vues des cellules sont issues d'une même classe abstraite, AbstractCellView. AbstractCellView, EdgeView et VertexView sont des classes de JGaph.

B. Implémentation de l'algorithme de placement

L'algorithme de placement est implémenté en Java dans la classe MassSpringLayout. Elle se compose de différentes méthodes d'initialisation pour le placement des nœuds, d'une méthode RepulsionSystem() implémentant le système physique d'interaction électromagnétique, d'une méthode MassSpringSystem() implémentant le système physique masses-ressort et diverses méthodes permettant de manipuler des vecteurs.

C. Exemple de représentation d'une voie métabolique avec PathwayBean

L'exemple porte sur un extrait de la voie métabolique du pentose phosphate. La Figure 21 illustre la représentation de cet extrait.

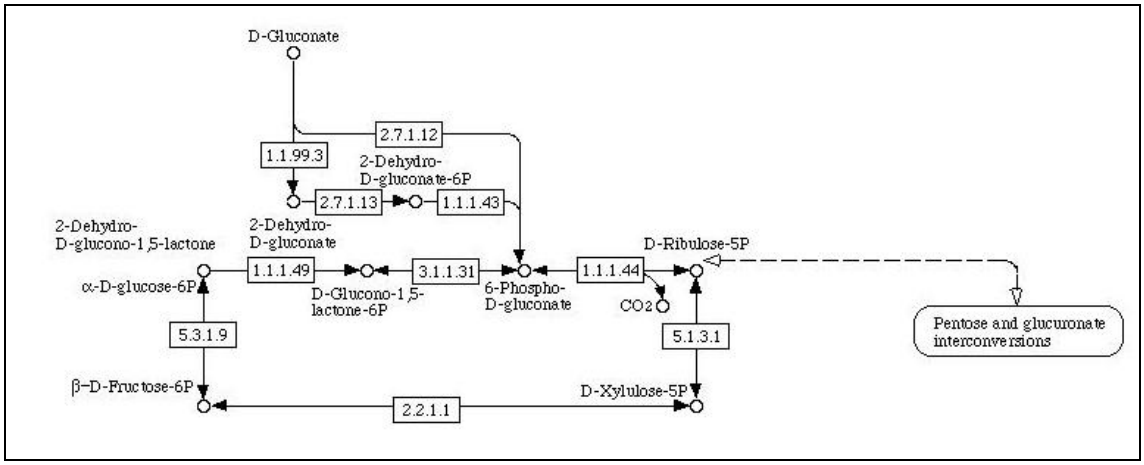


Figure 21 : Voie métabolique du pentose phosphate (KEGG)

Avant tout calcul de placement, les nœuds du graphe métabolique ont des positions aléatoires (Figure 22).

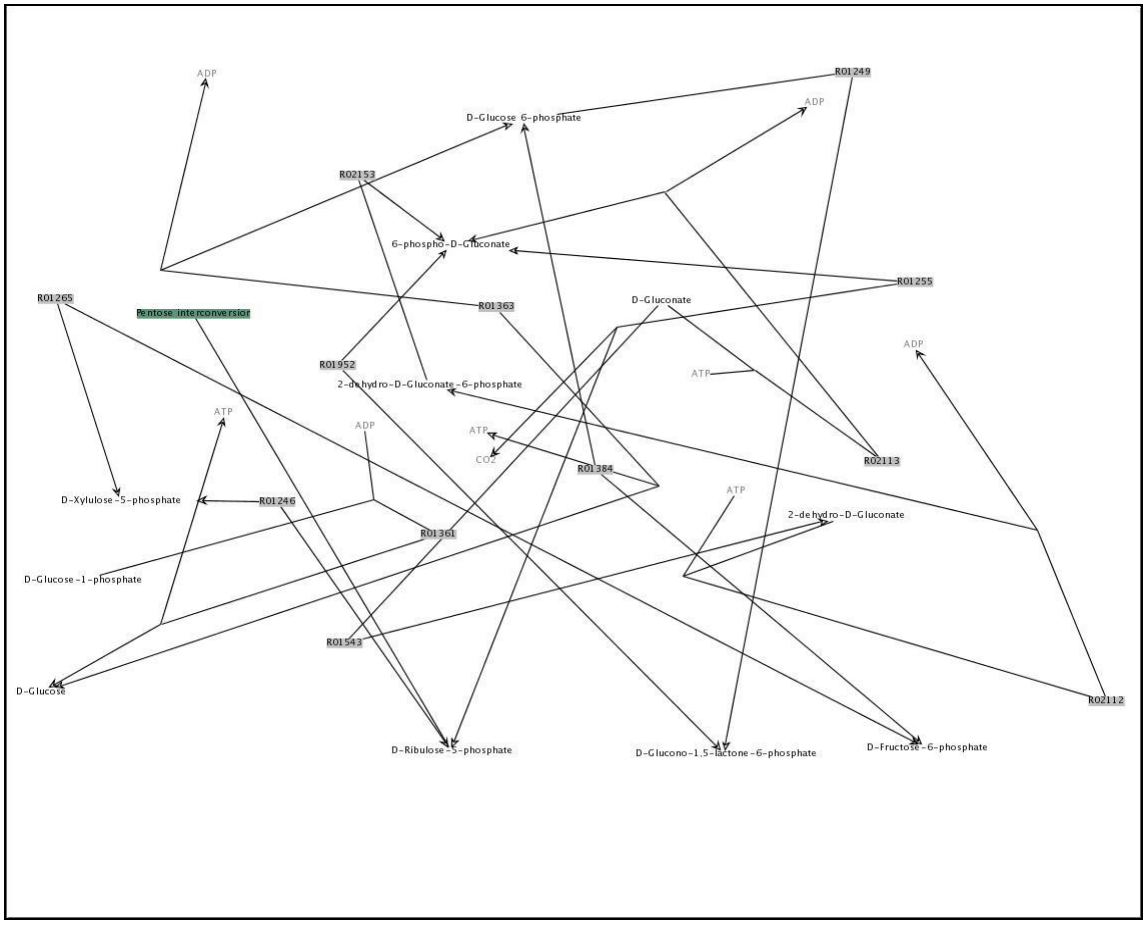


Figure 22 : Représentation du graphe métabolique avant l'application de l'algorithme de placement de PathwayBean.

Le calcul des positions des nœuds basé sur l'algorithme de placement par forces implémenté dans PathwayBean permet une représentation graphique claire et intuitive (Figure 23).

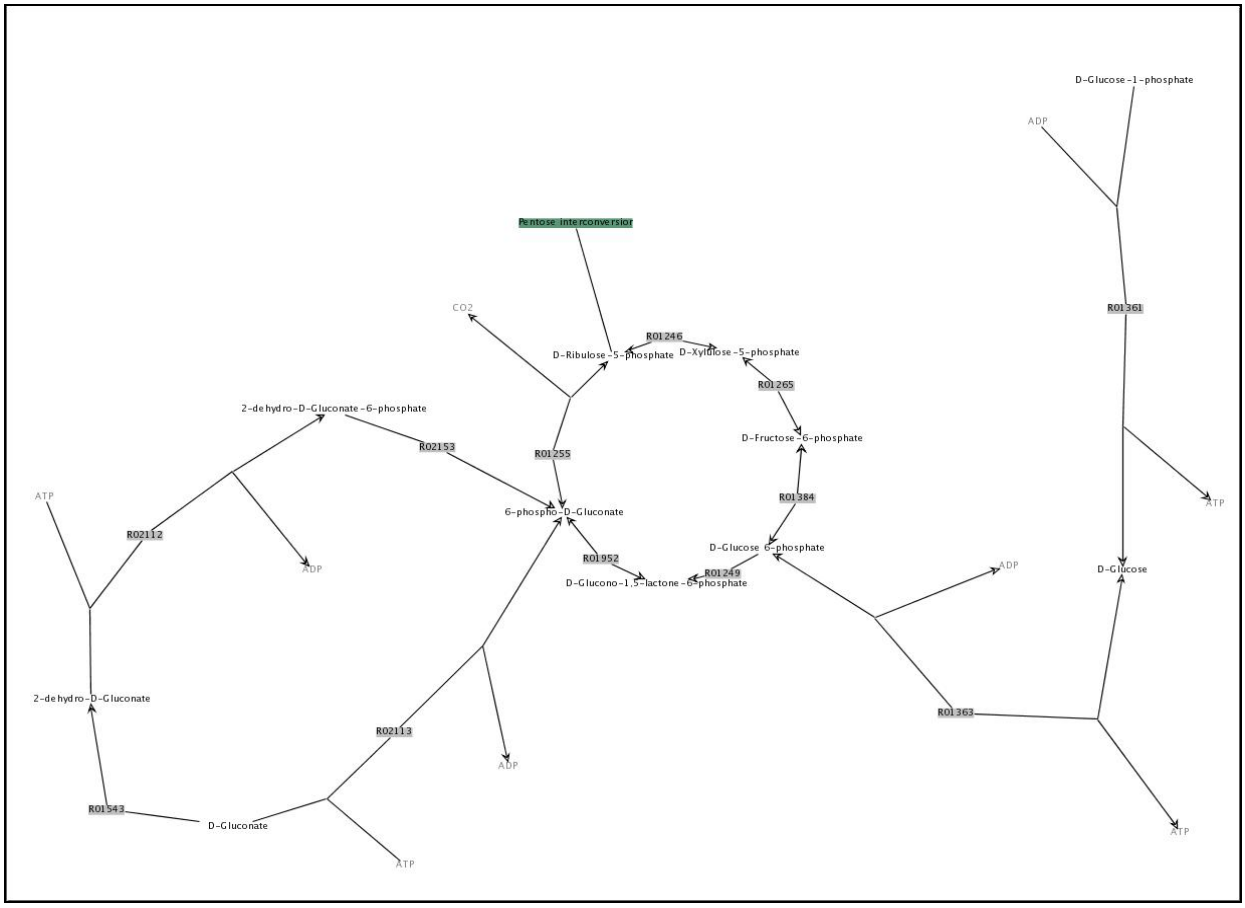


Figure 23 : Représentation du graphe métabolique après le calcul de placement.

L'information portée par un nœud du graphe peut être variée. Par exemple, il est intéressant de pouvoir afficher à la fois le nom et la formule développée d'un composé primaire. La figure 24 illustre l'intégration dans PathwayBean d'un autre composant graphique développé dans le cadre du projet Panoramix. Ce composant (MolBean) permet d'afficher la structure 2D d'un composé chimique.

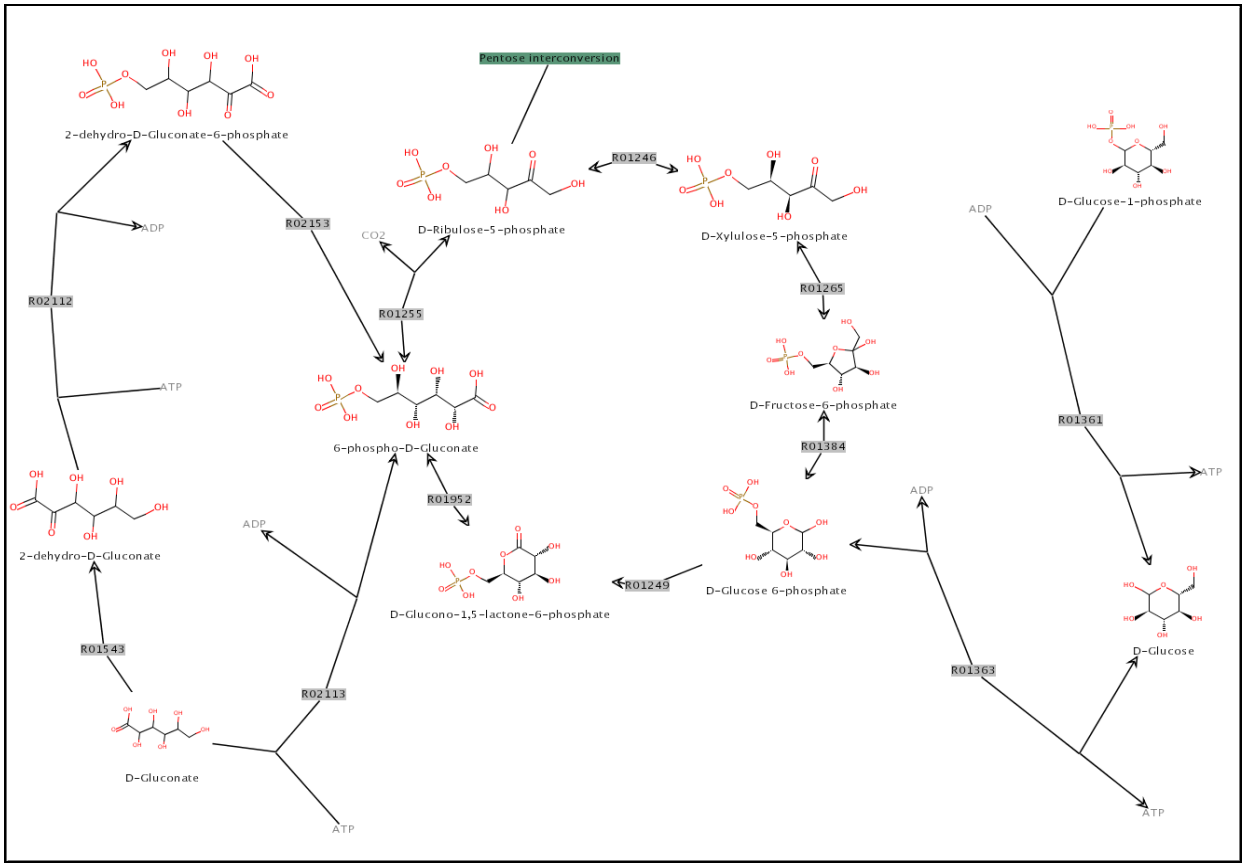


Figure 24 : Représentation du graphe métabolique (intégration de MolBean)

Conclusion et perspectives

En résumé, ce stage m'a permis de participer à un projet sur la totalité de son déroulement. J'ai travaillé sur toutes les phases de son développement : compréhension et spécification du projet, conception et modélisation, et implémentation. Ce projet a comporté les étapes suivantes:

- Définition d'un graphe métabolique
- Recherche de librairie pour la représentation d'un graphe
- Construction de l'architecture du composant PathwayBean
- Recherche et la conception d'un algorithme pour le placement des nœuds du graphe
- Implémentation des systèmes physiques propres à l'algorithme de placement

La conception et l'implémentation de l'algorithme de placement représentent les parties les plus importantes de mon travail. PathwayBean est un composant de représentation de voies métaboliques constitué d'un noyau dédié à la modélisation de graphe et d'un algorithme pour le placement des nœuds composant ce graphe. L'utilisation d'un algorithme basé sur un modèle physique permet une meilleure cohérence du placement automatique.

Quelques perspectives de développement sont à envisager afin d'optimiser le placement des nœuds :

- Respecter les normes de représentation des voies biochimiques. Par exemple, un nœud réaction pourra être représenté sur une droite rectiligne entre un nœud substrat et un nœud produit, et une arête reliant un nœud réaction à un nœud de composé secondaire pourra être représentée par une flèche courbée.
- Appliquer une rotation au motif.
- Appliquer une viscosité augmentant au fil du temps et généralisée à l'ensemble du graphe qui limitera les écarts de déplacement lorsque le nœud atteint sa position d'équilibre.
- Appliquer un mouvement brownien (agitation moléculaire) diminuant au fil du temps qui favorisera l'éloignement des nœuds trop proches au début du placement.

Ce stage a été très formateur et professionnalisant du point de vue des méthodes de travail de développement, de la maîtrise des outils utilisés dans l'équipe Hélix (Ant, CVS, XML,...), du travail en équipe, mais aussi du travail pluridisciplinaire. En effet, le thème de ce stage recouvrait différents domaines: la biochimie, la physique, l'algorithmique et le développement logiciel.

Références bibliographiques

Boehringer Biochemical Pathways poster
Gerhard Michal, Biochemical Pathways, Boehringer Mannheim, 1993

G. di Battista, P. Eades, R . Tamassia and I.G. Tollis
Graph drawing.
Prentice Hall, 1999

T. Fruchterman and E. Reingold
Graph drawing by force-directed placement.
Soft. Pract. And Exp., vol. 21(11) : 1129-1164 , 1991

W.T.Tutte (1963)
How to draw a graph.
Proceedings London Mathematical Society, vol. 13(3) :743-76

F. Schreiber (2002)
High quality visualization of biochemical pathways in BioPath
In Silico Biology 2, 0006, 2002

F.J. Brandenburg, M. Forster, A. Pick, M. Raitner and F. Schreiber (2001)
BioPath, Visualization of Biochemical Pathways.
Proceedings of the German Conference on Bioinformatics (GCB'01)

P. Karp and S. Paley (1994)
"Representations of metabolic knowledge: Pathways"
in Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology (Altman, R. and Brutlag, D. and Karp, P. and Lathrop, R. and Searls, D. eds.)

Ogata H., Goto S., Sato K., Fujibuchi W., Bono H. and Kanehisa M.
KEGG: Kyoto Encyclopedia of Genes and Genomes.
Nucleic Acids Research, 27, 29-34 (1999).

Isabel Rojas, Luca Bernardi, Esther Ratsch, Renate Kania, Ulrike Wittig and Jasmin Saric
A database system for the analysis of biochemical pathways, 2002.

ANNEXE : le projet Panoramix

Knowledge Bases for Genomics

We are currently developing three knowledge bases dedicated to bacterial genomes annotation. Each of them deals with a particular aspect of genomic and post-genomic data :

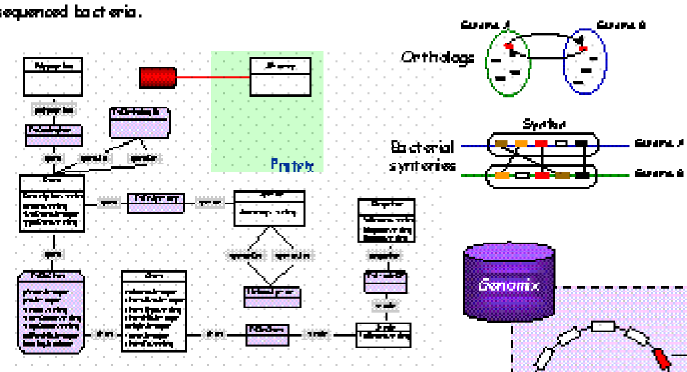
- Genomix
- Proteix
- Metabolix

} AROM bases

AROM is an Object Based Knowledge Representation System developped at INRIA Rhône-Alpes (http://www.inrialpes.fr/romans/pub/arom/).

Knowledge representation model based on classes and associations (n-ary).
AML : Algebraic Modeling Language to perform complex queries.

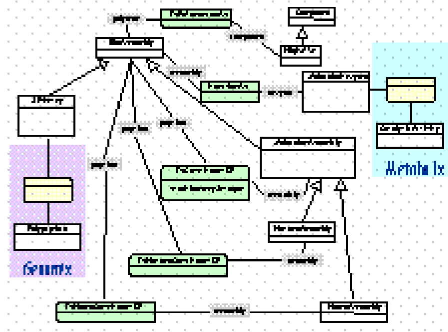
Genomix concerns organisms and their genes with a special emphasis to completely sequenced bacteria.



Data
- NCBI
  37 microbial genomes

Proteix deals with the product of the genes (proteins) with a special emphasis to Molecular enzymes.

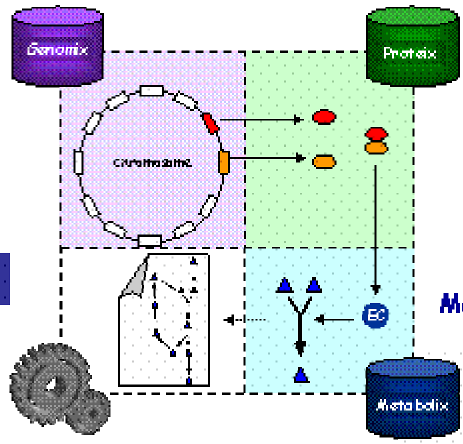
Data
- SwissProt (Expasy) complete proteomes
- Enzyme (Expasy) Repository of information on the nomenclature of enzymes (EC numbers)



Enzyme: EC number + reaction + polypeptide

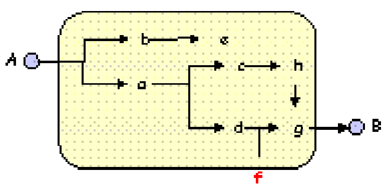
gene - protein - enzyme not always 1:1:1 relationship

2 concepts:
CatalyticActivity : Enzyme classification (EC number)
MolecularEnzyme : MolecularAssembly catalyzing a biochemical reaction



Metabolic pathways Reconstruction

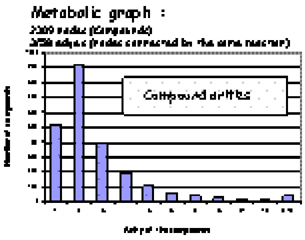
Goal : ab-initio reconstruction of biochemical pathways



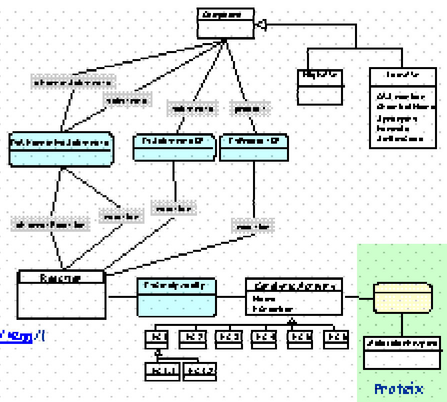
Formulation :
Minimum cost paths in a graph between two nodes

Constraints :
- Number of reactions
- Compounds
  - internal / external
  - energetic
- Energetic balance (ATP NADH ...)
- Chromosomal distance between the genes coding for the enzymes involved in the reconstructed pathway

Metabolic graph :



Metabolix is dedicated to intermediary metabolism.



Actors
- Compounds (low MW: glucose, NAD, ... high MW: protein ...)
- Reactions
- Enzymes (CatalyticActivity)

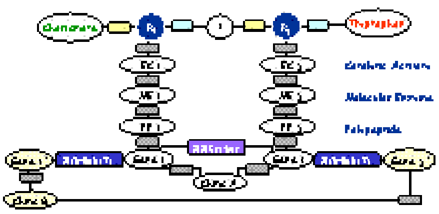
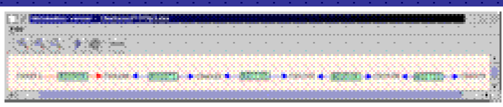
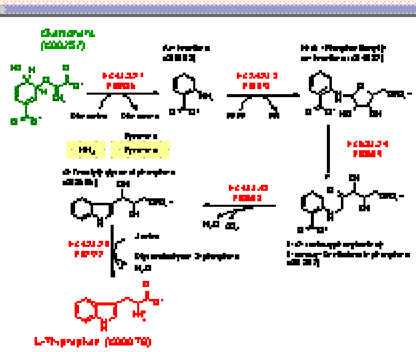
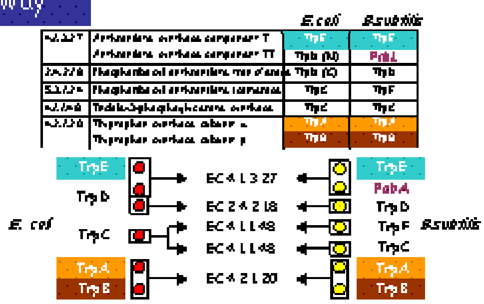
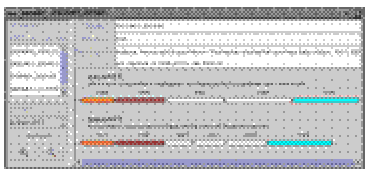
Data
- KEGG
  (http://www.genome.ad.jp/kegg/)
  3 849 compounds
  3 726 reactions
  3 514 EC numbers

Application : Tryptophan biosynthesis pathway

Q1: What is the set of reactions allowing transformation of Chorismate into Tryptophan ?

Q2: What are the enzymes in charge of these catalytic activities?

Q3: What are the polypeptide components of these enzymes ?

Q4: Where are located the genes coding for these polypeptides ? Is there any level of chromosomal organisation ?

Q5: Does this pathway exist in other bacteria ?