

EQUIPE CARTE

Réseaux de régulation de gènes

Initiation à la Recherche

Mickaël Charles & Romain Bailloux, encadrés par Emmanuel Jeandel

18/05/2015

Table des matières

Introduction.....	2
Contexte	2
Problématique.....	2
Sujet.....	2
Cadre du stage.....	2
Méthodologie de travail	3
Formalisme acquis.....	3
Architecture d'un réseau.....	3
Fonction locale de transition.....	4
Fonction de mise à jour	4
Graphe de transition	4
Formalisme construit	5
Coupe	6
Graphe de transition ouvert.....	6
Simplification.....	7
Opération « <i>papillon</i> ».....	7
Conclusion	8
Bibliographie.....	10
Outils utilisés	10

Introduction

Contexte

A l'orée du XXI^e siècle et de l'ordinateur quantique, il est important dans une société où la santé occupe une place importante d'augmenter l'espérance de vie des gens pour les faire travailler plus longtemps. C'est pourquoi les sciences de l'informatique et de la biologie se sont réunies pour former la bio-informatique, une nouvelle branche qui exploite toute la puissance des machines couplées aux méthodes formelles pour trouver des solutions pour dépister voire même guérir certaines maladies.

Problématique

Grace au réseau booléen nous pouvons représenter de manière simple des phénomènes complexes qui ont lieu entre des gènes et des protéines. La modélisation de ces systèmes se fait via un graphe où les sommets représentent les gènes et les arêtes orientées représentent les interactions (activation ou inhibition). On peut alors obtenir des propriétés particulières du vivant en observant la conformation du graphe.

Sujet

Proposé par Emmanuel Jeandel :

Le but du stage est d'étudier une méthode proposée par l'encadrant pour la manipulation des réseaux booléens, et en particulier pour capturer plus efficacement les comportements limites (les attracteurs), en introduisant un concept de modularité dans les réseaux booléens en régime synchrone ou asynchrone, interprété sous forme d'automate fini.

Cadre du stage

Durant notre Master 1 Informatique, nous avons eu l'occasion de participer lors d'une unité d'enseignement d'initiation à la recherche à l'avancement des recherches de ce domaine, à notre échelle infinitésimale.

Pour ce faire, nous avons commencé par lire de la documentation sur les automates booléens¹, sur la régulation des gènes et de protéines² ainsi que notre ancien cours de 3^{ème} année de licence dispensé par monsieur E.Jeandel lui-même.

A l'issu de ces 4 mois de travail, nous avons pu expérimenter une méthode de simplification combinatoire permettant d'obtenir des résultats concrets sur de gros systèmes sans passer par des approximations heuristiques qui ont déjà fait leurs preuves mais qui ne garantissent pas la justesse ou l'optimalité du résultat obtenu.

¹ (Sené, 2012)

² (Fauré, Naldi, Chaouiya, & Thieffry, 2006)

Méthodologie de travail

Afin de nous informer sur le sujet et de prendre connaissance de son cadre, nous avons pris rendez-vous avec notre encadrant. Lors des premières rencontres, il nous a introduits au sujet en présentant les réseaux booléens, la définition de graphe de transitions et d'attracteurs.

Il nous a ensuite parlé de sa méthode de construction d'un graphe de transition ouvert, de sa simplification et de l'opérateur papillon qui permet la fusion de deux instances pour obtenir un graphe de transition « fermé ».

A partir de cette base de définitions formelles, nous avons posé sur papiers des théorèmes et leurs démonstrations pour tenter de prouver la complétude et la décidabilité de cette méthode.

La réussite de cette étape théorique nous a mené à implémenter cette solution en Java, et à mesurer son efficacité sur des graphes construits à la main comme sur des réseaux de régulations réels.

Formalisme acquis

Architecture d'un réseau

Nous allons définir ce qu'est un réseau booléen :

C'est un graphe orienté $G = (V, A)$ où V est l'ensemble des sommets (numéroté de 0 à $n-1$) et A est l'ensemble des interactions entre les sommets.

On définit les voisins sortants du sommet i par $\Gamma_G^+(i) = \{j \mid (i, j) \in A\} \subseteq V$

On définit le degré sortant de i par $\deg_G^+ = |\Gamma_G^+(i)|$

On définit les voisins entrants du sommet i par $\Gamma_G^-(i) = \{j \mid (j, i) \in A\} \subseteq V$

On définit le degré entrant de i par $\deg_G^- = |\Gamma_G^-(i)|$

On définit un chemin de i_0 vers i_p par une liste de sommet (i_0, \dots, i_p) tel que $\forall k \in (0..p-1)$ on a $(i_k, i_{k+1}) \in A$

Un chemin est *simple* s'il n'est pas un cycle. Un chemin est un cycle si $i_0 = i_p$ (ou bien un sommet apparait deux fois dans le chemin).

Une *composante fortement connexe* est un sous-graphe G' de G où pour tout couple de sommet de G' , il existe un chemin de l'un vers l'autre.

On définit une *configuration* (ou état) du réseau booléen par :

$$x = (x_0 \dots x_{n-1}) \in \{0,1\}^n \text{ telle que } x_i \text{ est la valeur (ou état) du sommet } i.$$

Fonction locale de transition

Pour préciser la nature des interactions dans le réseau booléen, nous définissons pour chaque sommet i une fonction locale de transition $f_i : \{0,1\}^n \rightarrow \{0,1\}$ telle que :

$$\forall i \in V, \exists j \in V, \exists x \in \{0,1\}^n, f_i(x) \neq f_i(\bar{x}^j) \Leftrightarrow (j, i) \in A$$

Ainsi, comme chaque fonction locale de transition définit les interactions entre sommet, alors on peut entièrement définir un réseau booléen grâce à un ensemble de n fonctions locales.

$$\begin{cases} f_0(x) = x_1 \wedge x_2 \\ f_1(x) = x_0 \vee \neg x_2 \\ f_2(x) = \neg x_0 \end{cases}$$

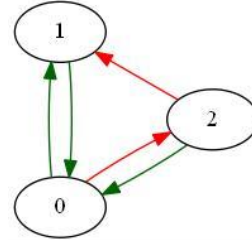


Figure 1 : Réseau correspondant

Fonction de mise à jour

Le passage d'une configuration à une autre est un événement qui correspond à la mise d'un ou plusieurs sommets (en appliquant leur fonction locale de transition). Si tous les sommets sont mis à jour en même temps, on parle de mise à jour synchrone. Dans le cadre de nos recherches, nous ne considérons que les *misés à jour asynchrone*, c'est-à-dire qu'on changera de configuration par l'application d'une seule fonction de transition. Cet événement est dit *atomique*.

La fonction de mise à jour d'écrit $F_i : \{0,1\}^n \rightarrow \{0,1\}^n$ telle que :

$$\forall x \in \{0,1\}^n, F_i(x) = (x_0, \dots, x_{i-1}, f_i(x), x_{i+1}, \dots, x_{n-1})$$

Le tableau ci-dessous montre les différents changements d'état du réseau de la figure 1 en fonction du choix du sommet mis à jour.

$x = (x_0, x_1, x_2)$			$f_0(x)$			$f_1(x)$			$f_2(x)$		
0	0	0	0	0	0	0	1	0	0	0	1
0	0	1	0	0	1	0	0	1	0	0	1
0	1	0	0	1	0	0	1	0	0	1	1
0	1	1	1	1	1	0	0	1	0	1	1
1	0	0	0	0	0	1	1	0	1	0	0
1	0	1	0	0	1	1	1	1	1	0	0
1	1	0	0	1	0	1	1	0	1	1	0
1	1	1	1	1	1	1	1	1	1	1	0

Figure 2 : Changement d'état en fonction du sommet mis à jour

Graphe de transition

Toutes ces configurations nous permettent de construire un graphe de transition. Celui-ci est défini par un graphe orienté $\mathcal{G} = (B, T)$ où l'ensemble des sommets sont les configurations ($B \in \{0,1\}^n$) du réseau et T est l'ensemble de transition (ici asynchrone) qui font passer le réseau d'un état à un autre. On peut aussi le définir formellement de la manière suivante :

$$x \rightarrow y \Leftrightarrow \exists i \in A \begin{cases} \forall j \neq i \ y_j = x_j \\ \text{sinon } y_i = f_i(x_k)_{k \in \Gamma^-(i)} \end{cases}$$

Le graphe de transition ci-dessous est associé au réseau booléen de la figure 1.

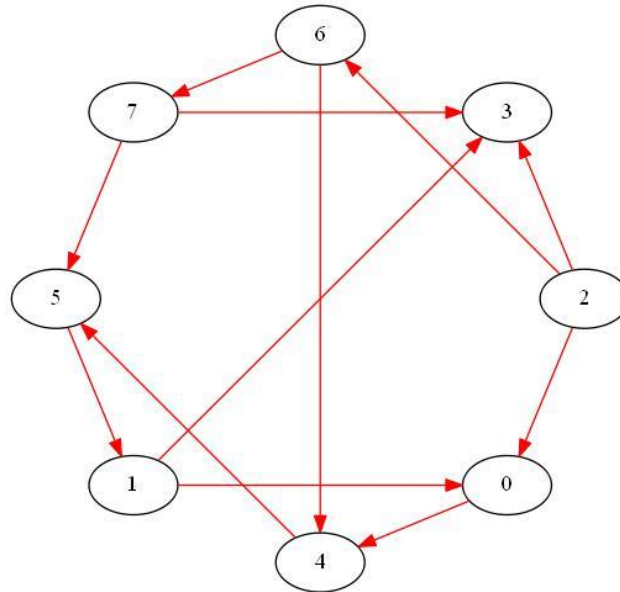


Figure 3 : Graphe de transition

Les comportements du vivant peuvent être détectés en observant les points fixes et attracteurs du graphe de transition.

On définit un point fixe par : $x \mid x \rightarrow y \Rightarrow y = x$

et un attracteur par : $x \mid x \rightarrow^* y \Rightarrow y \rightarrow^* x$

Notation : On dira que $x \rightarrow^* y$ s'il y a un chemin de x jusqu'à y.

Formalisme construit

Pour un réseau booléen de n sommets, on obtient un graphe de transition de 2^n sommets. La taille du graphe d'interaction augmente et celle du graphe de transition explose de manière exponentielle, nous avons donc travaillé sur une méthode proposée par E. Jeandel pour trouver leurs particularités sans avoir à faire tous les calculs.

Celle-ci se base sur le principe de « Diviser pour mieux régner » et consiste à *couper* le graphe d'interaction en deux, *calculer* les graphes de transition des deux parties, *simplifier* ces graphes par suppressions de sommets inutiles (qui n'apparaîtront jamais dans un attracteur) et à *fusionner* les graphes résultants de ces modifications pour obtenir le graphe de transition simplifié du système, sans perte d'informations utiles.

Cette méthode mise au point par notre encadrant au cours de ses recherches nécessite de préalablement poser des définitions permettant de démontrer quelques théorèmes.

Coupe

On définit la coupe de l'automate $A = B \uplus C$ de telle sorte qu'il y a une arête ou plus entre B et C, toutes de même source : noté c pour celles allant de C vers B et b pour l'autre sens.

On définit $B' = \{b'_k \mid k \in [0; l[]$ l'ensemble des sommets d'arrivée des arêtes issues de c et $C' = \{c'_k \mid k \in [0; n - l[]$ les sommets d'arrivée des arêtes issues de b .

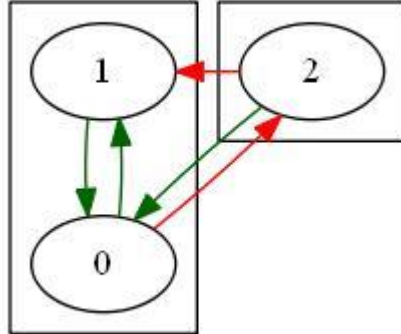


Figure 4 : Coupe du graphe de la figure 1

Graphe de transition ouvert

Le graphe ouvert est le graphe de transition issue des configurations possibles de B, pouvant recevoir des informations de l'extérieur pour ses sommets frontière (ici B'). On définit donc deux types d'arêtes, les **vertes** qui indiquent que B envoie la valeur de son sommet b et ne change pas de configuration, et les **rouges** qui indiquent la réception de la valeur de c entraînant un changement de configuration. On annotera ces flèches par * lorsque les sommets qui changent n'appartiennent pas à B' car la valeur de c ne sera pas utilisée.

Il y aura donc un sommet par configuration, et des arêtes définies par :

$$x \xrightarrow{x_b} x$$

$$x \xrightarrow{s} y \mid s \in \{0,1\} \text{ et } \exists i \text{ tel que}$$

$$\begin{cases} \text{Si } i \neq b'_k \left\{ \begin{array}{l} \text{Si } j \neq i \ x_j = y_j \\ \text{Sinon } y_j = f_i((X_k)_{k \in \Gamma^-(i)}) \end{array} \right. \\ \text{Sinon} \left\{ \begin{array}{l} \text{Si } j \neq i \ x_j = y_j \\ \text{Sinon } y_j = f_i((X_k)_{k \in \Gamma^-(i) \cup \{c\}}) \end{array} \right. \end{cases} \quad \text{avec } 0 \leq k \leq l$$

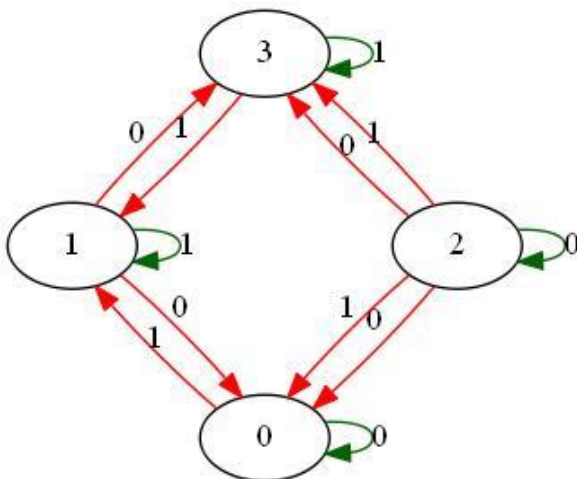


Figure 5 : Graphe de transition ouvert de la partie gauche de la coupe

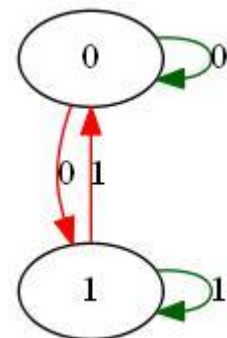


Figure 6 : de la partie droite

Simplification

Après avoir obtenu les graphes de transitions ouverts, nous avons remarqué que certains sommets respectent des propriétés qui permettent de dire que ceux-ci ne feront jamais partie d'un attracteur ou point fixe.

Par exemple si un sommet pointe un autre via une $\xrightarrow{*}$ et qu'il n'existe pas de retour vers lui, alors ce sommet ne peut pas faire partie d'un attracteur car si on le quitte, on ne le retrouvera plus jamais, et peut donc être supprimé.

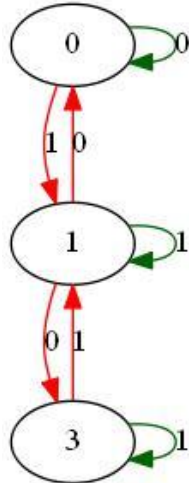


Figure 7 : Graphe de la figure 5 simplifiée

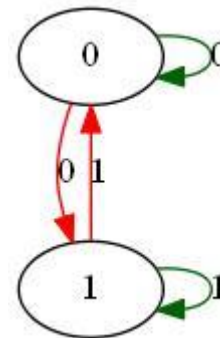


Figure 8 : Graphe de la figure 6 simplifiée

Comme on peut le voir, le sommet 2 de la figure 5 a été supprimé alors qu'aucune simplification n'a eu lieu sur la figure 6.

Opération « papillon »

Après avoir obtenu des graphes ouverts simplifiés nous avons défini un opérateur qui permet à partir de deux graphes de transitions ouverts d'obtenir un graphe de transitions fermé dans lequel les attracteurs seront mis en valeur.

Cet opérateur s'appuie sur le théorème suivant :

Théorème : Fusion

Le graphe G obtenu par la fusion des deux graphes ouverts de transitions B et C est défini par l'ensemble de sommets $\{ (x_B, x_C) \mid x_B \in B \text{ et } x_C \in C \}$ et l'ensemble d'arcs définis par :

$$(x_B, x_C) \rightarrow (y_B, y_C) \Leftrightarrow \exists s \text{ tel que } \begin{cases} x_B \xrightarrow{s} y_B \text{ et } x_C \xrightarrow{s} y_C \\ x_B \xrightarrow{s} y_B \text{ et } x_C \xrightarrow{s} y_C \end{cases} \text{ ou}$$

On notera l'opérateur \bowtie tel que $G = (B \bowtie C)$.

Demonstration : On a $(x_B, x_C) \rightarrow (y_B, y_C)$, $y_C = x_C$ (resp. $y_B = x_B$)

$\exists i$ tel que :

cas 1 :

$$i \in B \setminus B' \text{ (resp. } i \in C \setminus C') \\ \forall j \neq i \ y_j = x_j \text{ et } y_i = f_i((x_k)_{k \in \Gamma^-(i)})$$

cas 2 :

$$i \in B' \text{ (resp. } i \in C') \\ \exists \beta \text{ tel que } i = b'_\beta \text{ (resp. } C'_\beta) \\ \forall j \neq b'_\beta \ y_j = x_j \text{ et } y_{b'_\beta} = f_{b'_\beta}((x_k)_{k \in \Gamma^-(b'_\beta) \cup \{C\}})$$

$$\begin{cases} x_C \xrightarrow{x_C} y_C \\ x_B \xrightarrow{x_C} y_B \end{cases} \text{ resp. } \begin{cases} x_B \xrightarrow{x_B} y_B \\ x_C \xrightarrow{x_B} y_C \end{cases}$$

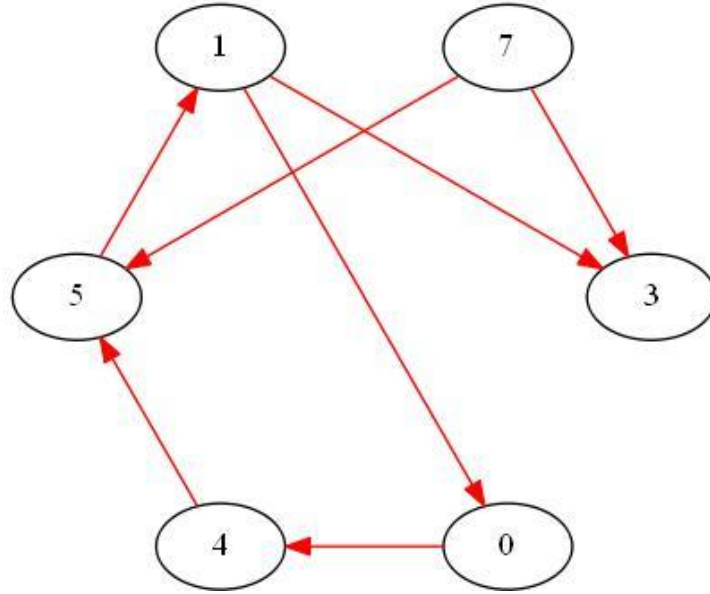


Figure 9 : Résultat du papillon entre les graphes de transitions simplifiés

Conclusion

Notre implémentation éprouve quelques difficultés dans le cas d'interactions complexes comme le réseau modélisant le cycle cellulaire de la drosophile³ en figure 10 construit par l'équipe de A.Fauré grâce à l'outil Ginsim. En effet ce réseau booléen ne peut pas (d'après l'exécution infinie de notre programme) trouver une coupe respectant les conditions définies précédemment. Cela ne veut pas dire que la méthode est un échec puisque nous ne l'avons pas testé exhaustivement sur un panel suffisamment large de réseaux. Le formalisme sur papier montre qu'il ne faut pas abandonner cette méthode prometteuse. Nos travaux restent quand même une bonne base de travail pour des projets futurs.

³ (Fauré, Naldi, Chaouiya, & Thieffry, 2006)

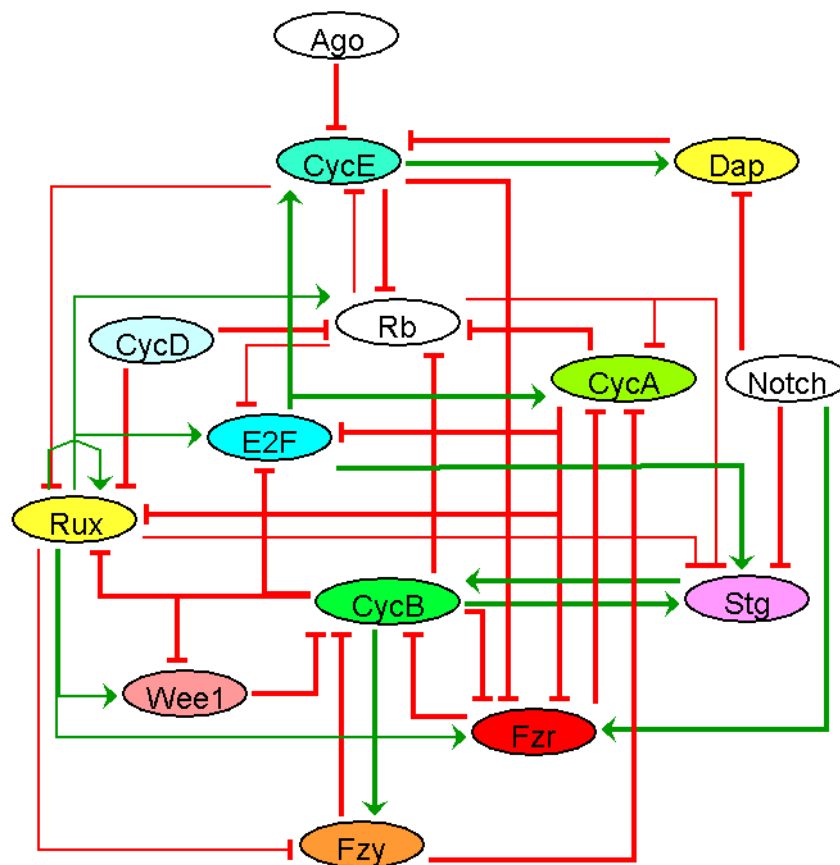
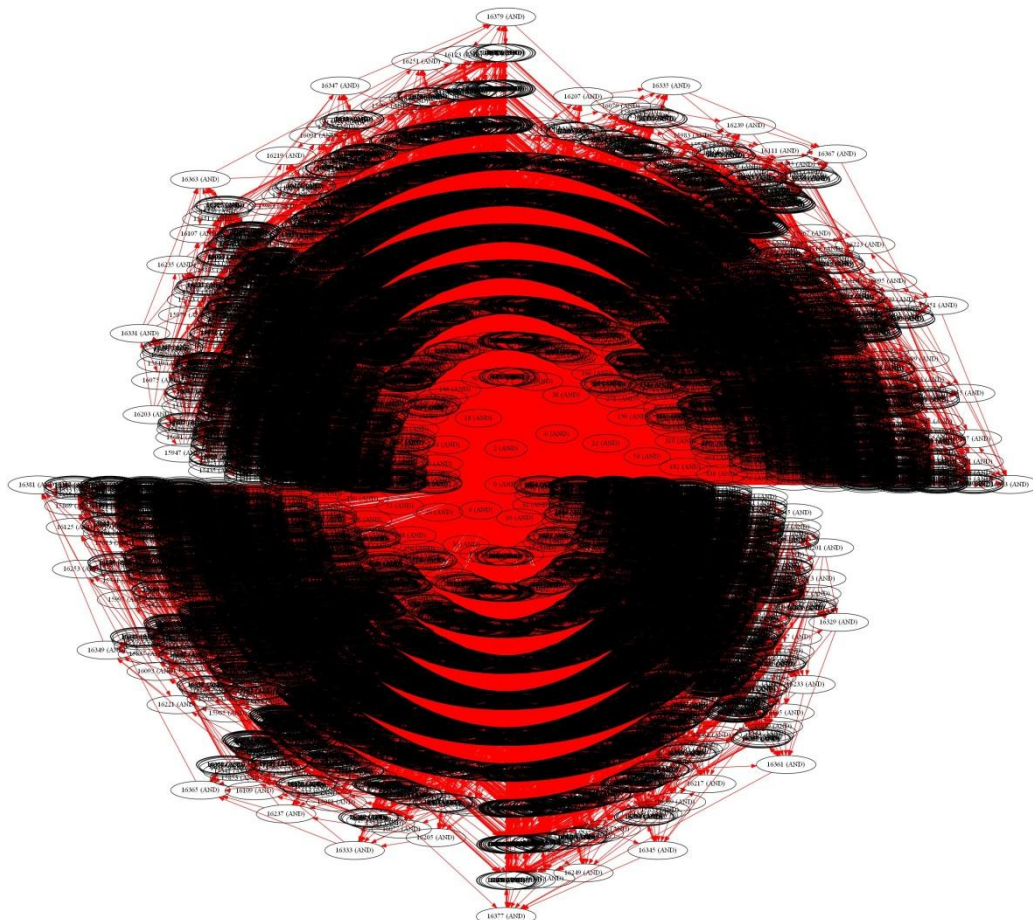


Figure 5 : Graphe d'interaction des gènes de la drosophile



Comme on peut le voir ci-dessus, pour seulement un réseau de 14 sommets, le graphe de transition est illisible. A partir de 30 sommets, ce n'est plus calculable de manière aussi brute.

Bibliographie

Fauré, A., & Chaouiya, C. (2009). *Logical modelling of cell cycle control in eukaryotes : a comparative study*.

Fauré, A., Naldi, A., Chaouiya, C., & Thieffry, D. (2006). *Dynamical analysis of a generic Boolean model for the control of the mammalian cell cycle*. Marseille: Campus scientifique de Luminy CNRS.

Naldi, A., Carneiro, J., Chaouiya, C., & Thieffry, D. (2010). *Diversity and Plasticity of The Cell Types Predicted from Regulatory Network Modelling*.

Sené, S. (2012). *Sur la bio-informatique des réseaux d'automates*. *Discrete Mathematics*. Université d'Evry-Val d'Essonne.

Outils utilisés

- <http://ginsim.org> – Outils de modélisation de réseau de régulation
- <https://eclipse.org> – Environnement de développement
- <http://graphviz.org> – Outils de génération de graphe