

Travail mi-juillet

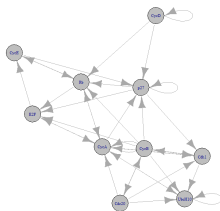
Zachary ASSOUMANI

26 juillet 2023

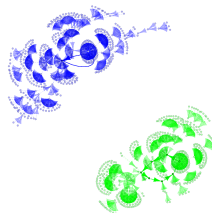
Prise en main de BoolNet

Outils d'analyse de BN et de PBN

- Calcul d'attracteurs, simulation, table de transition...
- Exemples : *cellcycle*, *examplePBN*



Graphe de régulation.



STG.

Classe PBN : présentation

Paramètres

- n (*int*) : nombre de gènes
- *indep* (*bool*) : marginales indépendantes ou non
- f (*array*) : fonctions de transition. Si *indep* : classes de fonctions $f = F_1 \times \dots \times F_n$. Si not *indep* : ensemble de contextes $f = f_1 \dots f_m$.
- c (*array*) : distribution des fonctions de transition. De même forme que f .
- *sync* (*bool*) : mise à jour synchrone ou asynchrone
- p, q (*float*) : facteurs de perturbation de gène et de changement de contexte

```
=====
PBN asynchrone (n=3 gènes)

Facteur de perturbation : p=0.00
Facteur de changement de contexte : q=1.00

Fonctions de transition :
x0 - 2 fonctions possibles
x1 - 1 fonctions possibles
x2 - 2 fonctions possibles

000 || 0 0 - 0 - 0 0
001 || 1 1 - 1 - 0 0
010 || 1 1 - 1 - 0 0
011 || 1 0 - 0 - 1 0
100 || 0 0 - 1 - 0 0
101 || 1 1 - 1 - 1 0
110 || 1 1 - 0 - 1 0
111 || 1 1 - 1 - 1 1
Distribution(s) de probabilité : [[0.6, 0.4], [1], [0.5, 0.5]]

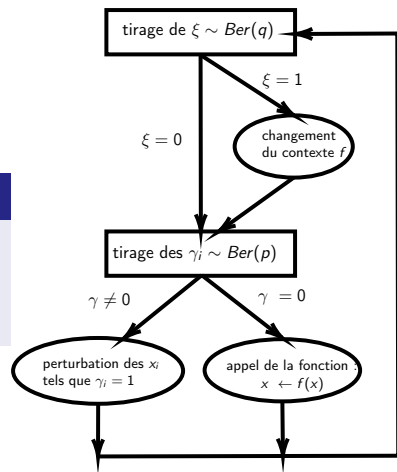
Etat actuel : [1, 0, 0]

Contexte actuel : f0_0, f1_0, f2_0
000 || 0 0 0
001 || 1 1 0
010 || 1 1 0
011 || 1 0 1
100 || 0 1 0
101 || 1 1 1
110 || 1 0 1
111 || 1 1 1
=====
```

Classe PBN : itération

Simulation : *step()*

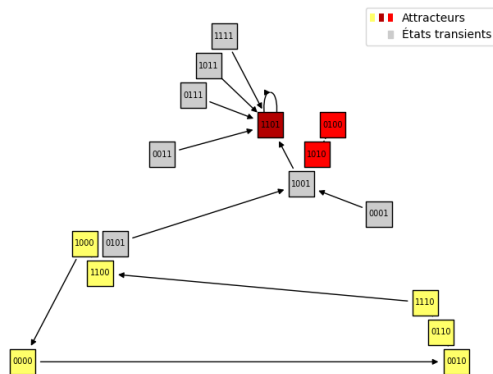
- changement de contexte : *switch()*
- perturbation des x_i : *flip()*
- appel de la fonction : *call()*



Classe PBN : STG

Construction du STG

- Construction de la liste d'adjacence
- Détection des attracteurs : SCC terminales
- Affichage



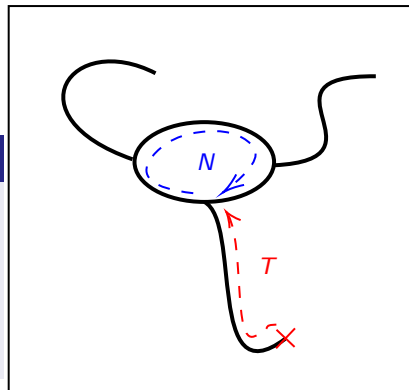
Classe PBN : loi stationnaire

`loi_stationnaire(T,N,R)`

R simulations :

- On pioche $x \in \{0, 1\}^n$
- On fait tourner T itérations
- On échantillonne les N suivantes

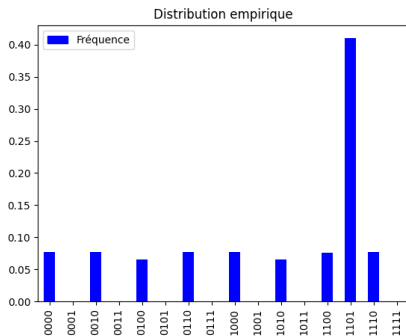
Affichage de l'histogramme.



Avec les T premières itérations, on espère atteindre des états récurrents.

Classe PBN : loi stationnaire

En synchrone :

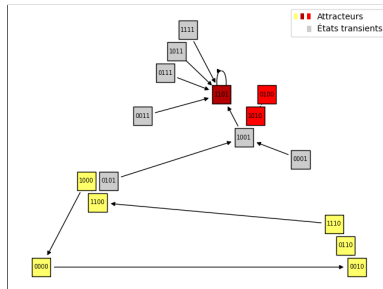
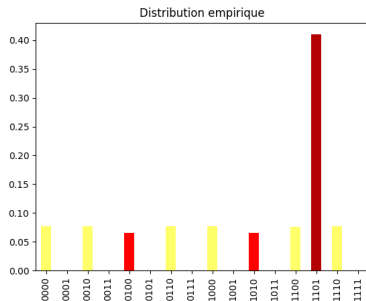


Exemple à $n = 4$ gènes (Abou-Jaoudé 2016)

Seule une fraction de l'espace d'états est échantillonnée à long terme : ici les états récurrents, c'est-à-dire l'union des attracteurs.

$$T > \max_A \max_{x \in B_A} \underbrace{\delta(x, A)}_{\text{distance de } x \text{ à son attracteur}}$$

Classe PBN : loi stationnaire



On remarque que pour un attracteur A donné, tous les états $x \in A$ ont la même probabilité stationnaire. . .

Classe PBN : loi stationnaire

Inférence des attracteurs synchrones

- Pour tout attracteur A , on a une probabilité $\frac{|B_A|}{2^n}$ de débuter une simulation dans son bassin B_A , et donc de finir par échantillonner A .
- Dans la phase d'échantillonnage, puisque l'attracteur est cyclique, on visite chaque $x \in A$ avec une proportion $\frac{1}{|A|}$.

→ Ainsi, la probabilité stationnaire d'un état $x \in A$ sera $S(x) \simeq \frac{|B_A|}{|A| \cdot 2^n}$.

Classe PBN : loi stationnaire

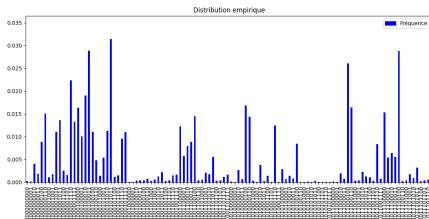
Inférence des attracteurs synchrones

En supposant que pour tous attracteurs A_1 et A_2 , $\frac{|B_{A_1}|}{|A_1|} \neq \frac{|B_{A_2}|}{|A_2|}$, on peut identifier les attracteurs en partitionnant les états visités par la hauteur de leurs barres dans l'histogramme.

- 1 Chaque taille $|A|$ d'attracteur s'infère comme le nombre de barres à une hauteur donnée.
- 2 La taille $|B_A|$ du bassin d'attraction associé à x s'approxime par $|B_A| \simeq S(x) \cdot |A| \cdot 2^n$.

Classe PBN : loi stationnaire

En asynchrone :



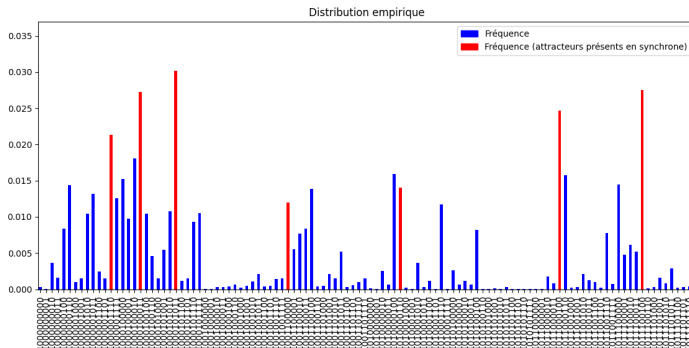
Exemple Th à $n = 10$ gènes (Fauré 2006)

La dynamique non-déterministe produit une loi stationnaire moins limpide. Dans cet exemple, le réseau asynchrone possède un attracteur complexe à 112 états et 338 transitions.

La présence de SCC non-terminales ne garantit pas, pour un T donné, que tous les états échantillonnés appartiennent à un attracteur.

Classe PBN : loi stationnaire

Remarque : dans cet exemple, on retrouve que les attracteurs synchrones (en rouge) sont parmi les plus visités en asynchrone.



Classe PBN : fonctions voisines dans le diagramme de Hasse

Curry 2019, section 6.2. Pour un ensemble de gènes i :

$$\begin{aligned} BN &\implies PBN \\ F_i = \{f_i\} &\implies F_i = \{f_i\} \cup \{\text{voisins de } f_i\} \end{aligned}$$

Les fonctions d'un gène comptent non seulement sa fonction de référence initiale, mais aussi les voisins de celle-ci dans le PO-Set des fonctions de régulation.

- Le modèle Th_23 étudié dans l'article a trois états stables phénotypes : Th0 - Th1 - Th2
- Six extensions possibles en PBN proposées
- Comparaison de la prévalence de chaque phénotype, en mise à jour synchrone comme asynchrone

Classe PBN : fonctions voisines dans le diagramme de Hasse

Import de fichiers

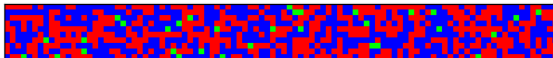
- Parsing d'un fichier .bnet pour créer un objet dans la classe "PBN".
- Pour $R = 1000$ simulations, on consigne l'état stable d'arrivée.



Table0_p08.bnet (synchrone)



Table0_p08.bnet (asynchrone)



Exemple avec le BN initial. 7.7 / 55.8 / 36.5 - 3.2 / 48.5 / 48.3

Classe PBN : fonctions voisines dans le diagramme de Hasse

Table1A_fall_d1_p08.bnet (synchrone)

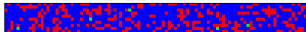


Table1A_fall_d1_p08.bnet (asynchrone)



0.5 / 28.6 / 70.9 - 9.5 / 40.7 / 49.8

Table1B_fall_d1_p08_siblings.bnet (synchrone)

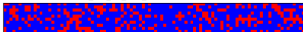


Table1B_fall_d1_p08_siblings.bnet (asynchrone)



0 / 28.9 / 71.1 - 8.7 / 43.5 / 47.8

Table1C_fgATA3_p08.bnet (synchrone)



Table1C_fgATA3_p08.bnet (asynchrone)



0 / 100 / 0 - 74.8 / 25.2 / 0

Table1D_ftbet_p08.bnet (synchrone)



Table1D_ftbet_p08.bnet (asynchrone)



22.9 / 77.1 / 0 - 88.1 / 11.9 / 0

Table1E_fil4_p08.bnet (synchrone)



Table1E_fil4_p08.bnet (asynchrone)



0 / 96.7 / 3.3 - 17.2 / 23.5 / 59.3

Table1F_fil4R_p08.bnet (synchrone)



Table1F_fil4R_p08.bnet (asynchrone)



0 / 74.0 / 26.0 - 14.4 / 23.6 / 62.0

Sur $R = 1000$ simulations : Th0 en vert / Th1 en rouge / Th2 en bleu.

À faire

- Détection des attracteurs avec les méthodes de BioLQM via Jython
- Loi stationnaire en synchrone : cas où deux attracteurs ont la même hauteur + intervalle de confiance sur la taille du bassin d'attraction en fonction de R
- Loi stationnaire en asynchrone : comparer les attracteurs asynchrones avec ceux de la classe de priorité (Fauré2006)
- Diagramme de Hasse : étudier les cas $q < 1$ et $p > 0$
- Générer de PBNs synthétiques pour généraliser les observations, dont celles sur les fonctions de voisines via l'algorithme de Pedro <https://github.com/ptgm/functionhood>