

# Module de recherche de motifs exceptionnels

S.Robin, M.Koskas, E.Birmelé, S.Schbath, G.Grasseau

NeMo

21-juin-2009

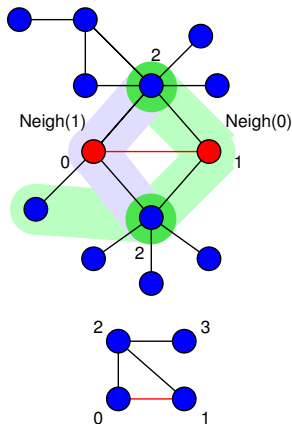
Dernière réunion, présentation :

- détermination de motifs exceptionnels,
- calcul de  $E^{mod}$ ,  $V^{mod}$  (graphes non-dirigés),
- prototype de recherche de motif et de calcul de  $E^{mod}$ ,  $V^{mod}$  (graphes non-dirigés).

Depuis :

- optimisation de la partie recherche de motif,
- graphes orientés et non-orientés,
- calcul de la probabilité critique,
- validation des résultats.

# Algorithme de *backtracking*



Étape du *backtracking*

Pour un motif fixé, la complexité est polynomiale (NP-complet) et même linéaire en fonction du nombre d'arêtes.

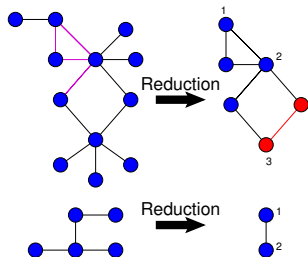
→ implémentation au plus linéaire.

Étape  $k$  :

- Cherche tous les noeuds dans le graphe connectés de la même façon (*subgraph isomorphism*) que le  $k^{i\text{eme}}$  noeud du motif.
- réitère l'opération (étape  $k + 1$ ) pour chaque noeud trouvé
- L'opération la plus coûteuse, implémentation par un *tas trié*.

Contrainte pour la mat. d'adjacence du motif  
tout noeud  $i$  doit être connecté avec au moins un des  $i - 1$  premiers noeuds

# Algorithme : réduction des motifs



Réduction graphe et motif.

$$\text{Ex. : } N_{\text{occ}} = \binom{4}{2} \times \binom{1}{1}$$

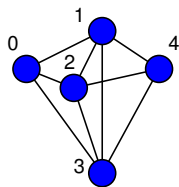
→ très efficace mais problème avec les noeuds qui ne sont pas de degré 1 (voir figure)!

Idée : simplifier l'arbre de recherche en supprimant les noeuds "inutiles" des  $\mathbf{m}$  et  $\mathbf{G}$  : noeuds de degré 1.

Étapes :

- Graphe (réseau et motif) initial → un graphe valué par le nombre de voisin de degré 1.
- recherche du motif réduit dans le graphe réduit,
- Si occurrence alors par combinatoire on trouve le "vrai" nombre d'occurrences du motif initial.

# Classes topologiques et isomorphisme

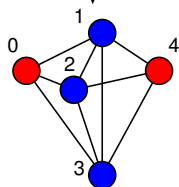


Recherche

Perm. possibles

0	1	2	3	4
4	23	13	12	0

C.T.



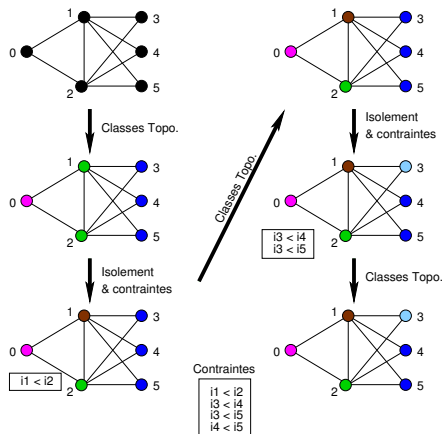
Classes topologiques :

- Utilisation de l'algo. de recherche (sans contrainte) du motif sur lui-même. On obtiendra toutes les permutations  $aut(\mathbf{m})$ .

Isomorphisme :

- Recherche (sans contraintes) d'un motif  $m_1$  dans un motif  $m_2$  (nombre d'arêtes identiques).

# Backtracking avec contraintes



Détermination des contraintes.

Le problème :

- recherche de motif à  $aut(\mathbf{m})$  près,
- éviter un tri parmi  $aut(\mathbf{m})$ .

Backtracking avec contraintes :

- Parcourir de façon unique le motif.

- Contraintes :

$$CT \rightarrow CT' + \text{contraintes}$$

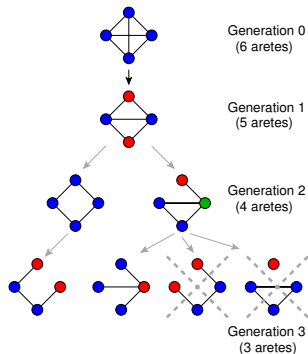
$$\text{avec } card(CT'(i)) = 1,$$

$j^{ieme}$  CT du motif

- extrait un noeud de la  $CT(i)$ , on rajoute une contrainte sur les indices du noeud  $j_1 < j_u, j_u \in CT(i), m > 1$

→ implémentation d'un algo. de recherche avec des motifs colorés (CT).

# Ensemble des motif à $k$ noeuds



*Génération de tous les motifs à 4 noeuds*

Trouver tous les motifs à  $k$  noeuds.

Étapes 0 : clique.

Étapes  $k$  :

- déterminer des CT,
- retirer une arête entre 2 noeuds parmi toutes les combinaisons de CT.
- vérifier si le motif est connexe,
- vérifier s'il n'est pas isomorphe à un motif de la même génération,
- passer à la génération suivante  $k + 1$ .

Extensions :

- Généralisation aux motifs orientés.
- Obtention des relations entre comptage motifs induits et non-induits

$$m_{wisk}^i = m_{wisk}^{ni} - 12 \times m_{clique}^{ni} - 4 \times m_{diamond}^{ni}$$

# Performances de la recherche de motif

- Comparaison du comptage avec Fanmod

Réseau PPI

*Saccharomyces cerevisiae*

Noeuds 4958, arêtes 17224,

$\langle \text{degree} \rangle = 3.4$

Nombre de noeuds	Fanmod (s)	Algo. (s)
4 (6)	27	8
5 (21)	2 071	372
6 (112)	> 157 909	28 462

→ efficace pour  $k \leq 7$ , ou sur des réseaux denses,

Réseau PPI

*Escherichia coli*

Noeuds 424, arêtes 519,

$\langle \text{degree} \rangle = 1.22$

Nombre de noeuds	Fanmod (s)	Algo. (s)
5 (21)	4	1,4
6 (112)	76	24
7 (853)	1 515	490
8 (11 117)	26 004	> 86 000

→ au-delà de 8 le nombre de motifs est trop important ( $\sim e^k$ ).

## Bilan :

- Prototype pour graphes orientés / non-orientés
- Modèle ER, EDD, Mixnet
- Calcul probabilités critiques
- Disponible fin juillet (peut fournir pour tester)

## État des développements logiciels NEMO :

- Package R d'un ss-ensemble de Motus (VL,SS). Forge ?
- Nouvelle version de `mixnet` en séparant les étapes : initialisation, étape E et étape M - forge INRA/mixnet/mixer-dev (VM).
- Interfacer `mixer` avec nouvelle version de `mixnet` (forge INRA)(GG)
- Nouvelle version évaluée `branches/MixWNet` (VM/FP).
- Recherche de motifs exceptionnels par concentration (EB/GG)
- Cohérence des dév. motifs topologiques/colorés (LB/GG).