Recherche d'un noyau dans un graphe aléatoire

Marco Illengo et Jean-Marie Le Bars

LMNO et GREYC, université de Caen

Journées Alea 2012

Plan de l'exposé

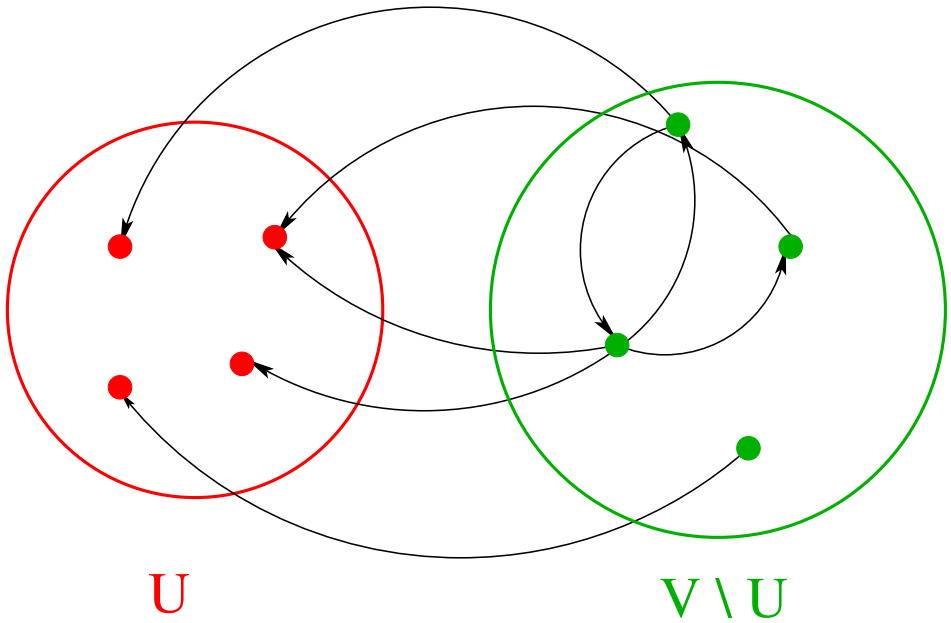
Plan de l'exposé

Définition de la propriété de noyau

Graphe orienté $D_n = \langle V_n, A \rangle$, $V_n = \{1, \ldots, n\}$, A ensemble d'arcs

Un noyau est un sous-ensemble U de V_n vérifiant les deux propriétés suivantes :

- U est un stable : pas d'arc entre deux sommets de U.
- K est dominant : pour tout sommet en dehors de U, il existe un arc vers un sommet du U.



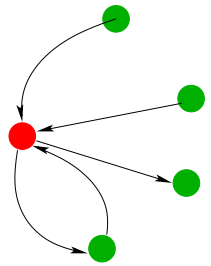
Définition de la propriété de noyau

ROUGE sommets du noyau

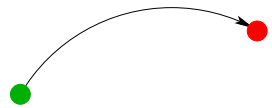
VERT sommets à l'extérieur du noyau

Contraintes locales

- tous les voisins d'un sommet rouge sont verts



- un sommet vert à au moins un sommet sortant rouge (témoin)



Recherche d'un noyau : coloriage des sommets en rouge et vert

Jeux combinatoires et graphes

Jeux à deux joueurs

Jeux finis à deux joueurs sans information cachée avec toujours un vainqueur.

Un des deux joueurs possède une stratégie gagnante

Graphe orienté

Sommets : configurations du jeu

Arcs : déplacements

Noyau = Ensemble des positions gagnantes

Celui qui possède une stratégie gagnante se déplace toujours vers un sommet du noyau.

Recherche d'un noyau dans un graphe

Problème de décision NOYAU

D_n vérifie NOYAU lorsqu'il possède au moins un noyau.

NOYAU est un problème NP-complet.

Notre objectif

Générer aléatoirement des instances difficiles.

Application en cryptographie

Cacher un noyau dans un graphe aléatoire.
(en collaboration avec Eleonora Guerini et Fabien Laguillaumie)

Utilisation des séries génératrices

On peut utiliser les séries génératrices lorsque le graphe est un arbre ou proche d'un arbre

Sur les arbres

- arbres planaires généraux
- arbres binaires
- arbres enracinés étiquetés

(voir l'article de Cyril Banderier, Markus Kuba, Alois Panholze, 2009)

Graphe avec un seul circuit

(Cyril Banderier, LB, Vlady Ravelomanana, 2004)

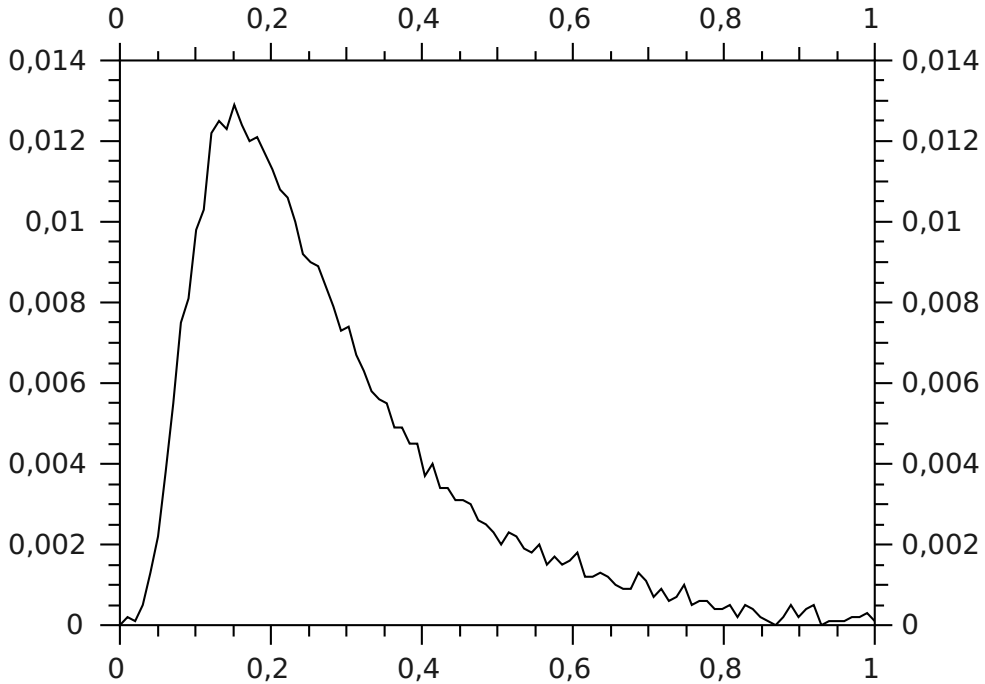
Graphes aléatoires

Modèle $\mathcal{D}(n, p)$

chaque paire de sommets est un arc avec probabilité p.

Complexité moyenne de l'algorithme $\mathcal{A}$

100 sommets et 100 itérations



Plan de l'exposé

Algorithme naïf $\mathcal{B}$

Taille des noyaux

On suppose que l'on connaît les tailles possibles : ensemble $\mathcal{I}$.

Recherche exhaustive

On effectue la recherche exhaustive sur tous les sous-ensembles de V_n de taille $r \in \mathcal{I}$.

Borne supérieure de la complexité

$$\sum_{r \in \mathcal{I}} \binom{n}{r} \; n \ln n.$$

Algorithme $\mathcal{A}$ – Étape 1

Algorithme pour trouver l'unique noyau d'un graphe sans circuit

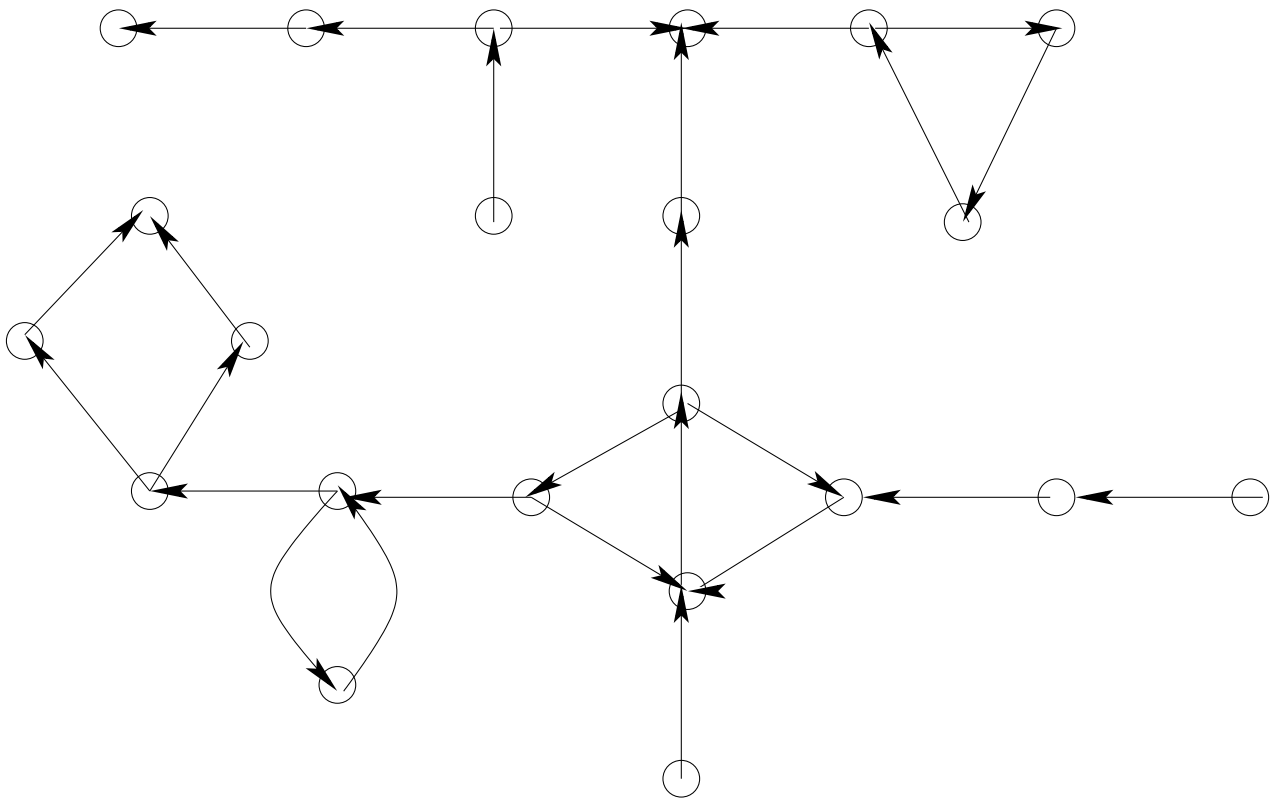
Coloriage à trois couleurs

- BLANC : sommets non traités
- ROUGE sommets dans le noyau
- VERT sommets à l'extérieur du noyau

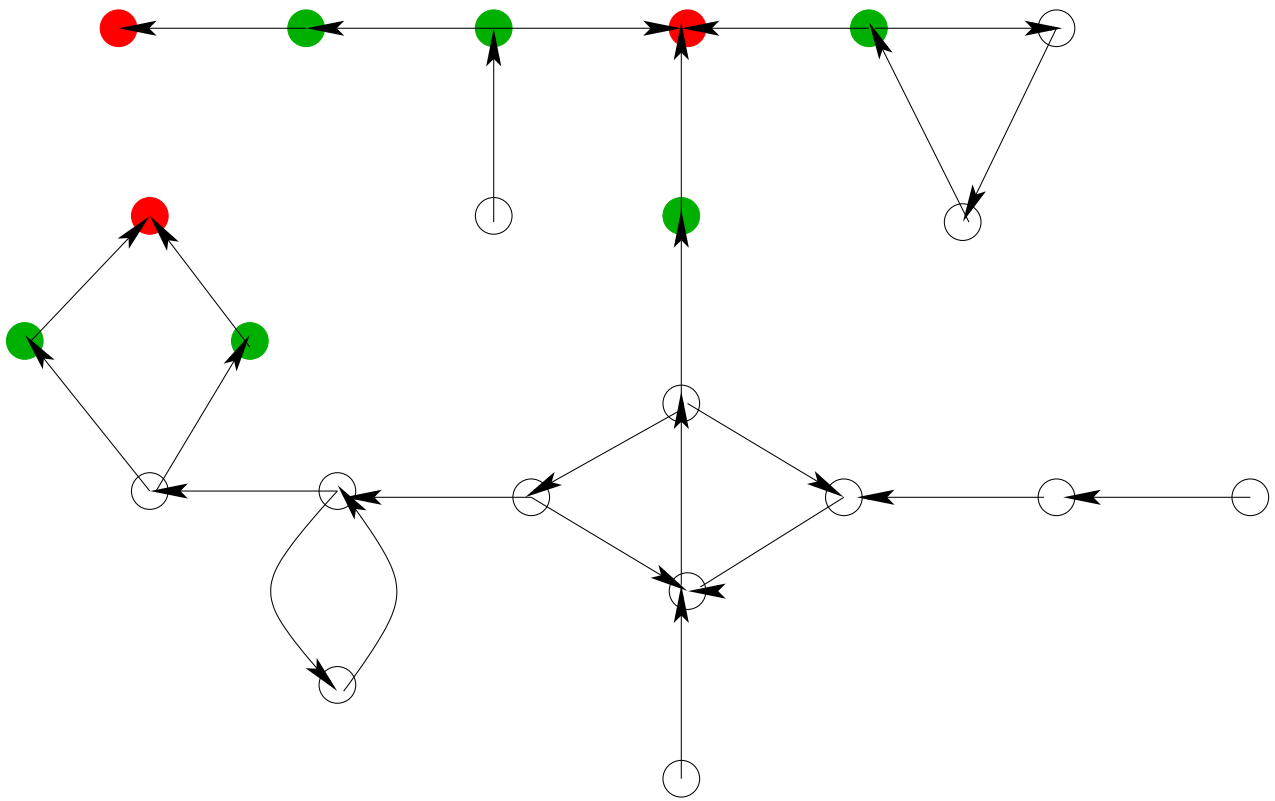
Algorithme $\mathcal{A}_1$

Tant que BLANC est non vide et qu'il reste un puits
    colorier les puits en rouge
    colorier les voisins des puits en vert
    supprimer du graphe les sommets coloriés

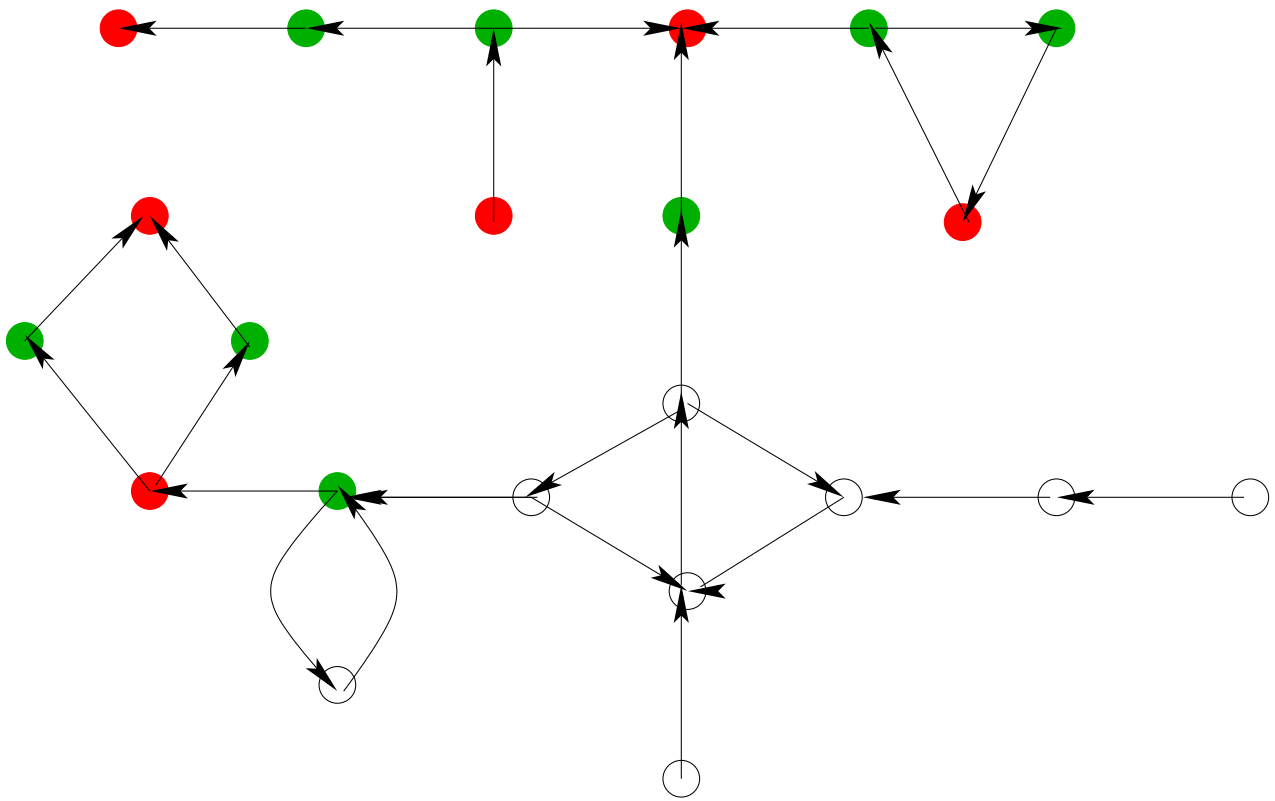
Algorithme $\mathcal{A}$ – Étape 1



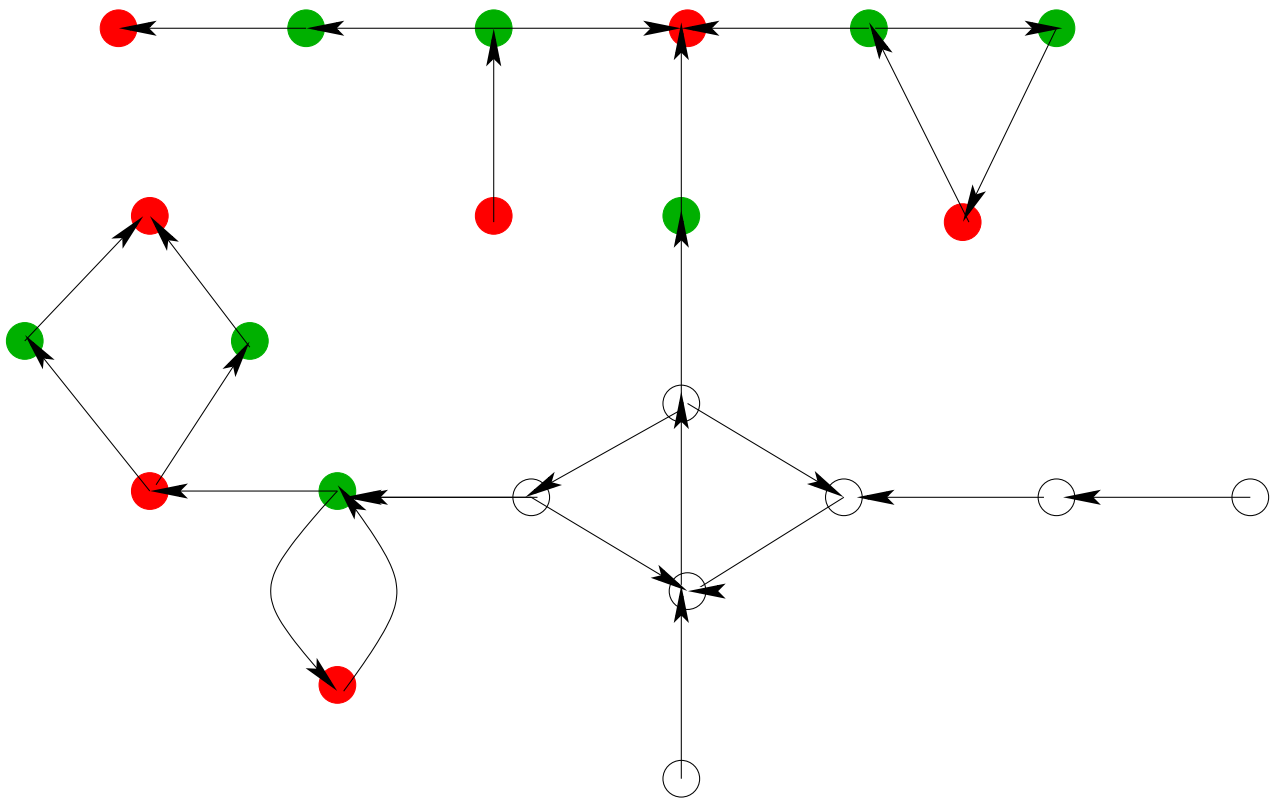
Algorithme $\mathcal{A}$ – Étape 1



Algorithme $\mathcal{A}$ – Étape 1



Algorithme $\mathcal{A}$ – Étape 1



Algorithme $\mathcal{A}$ – Étape 2

Backtracking simple, coloriage à quatre couleurs

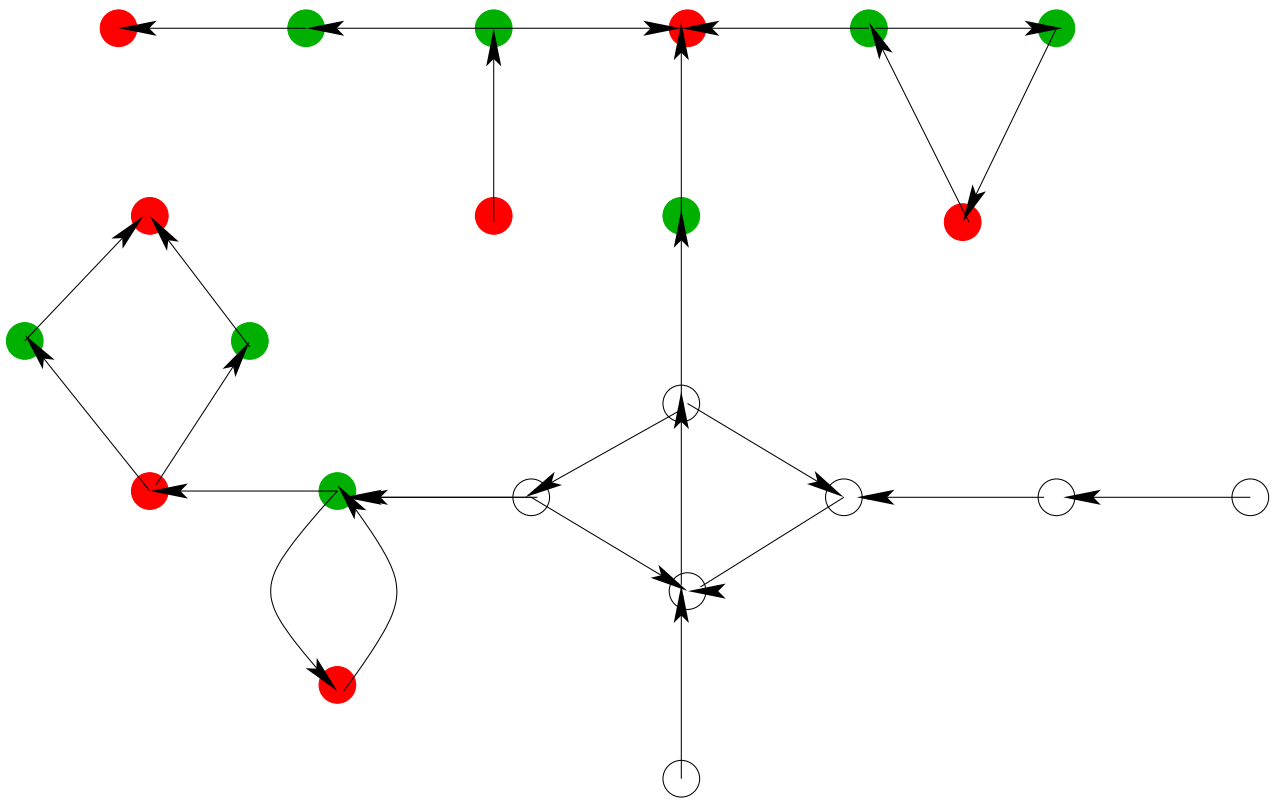
- BLANC sommets non traités
- ROUGE sommets dans le noyau
- BLEU sommets à l'extérieur du noyau sans témoin
- VERT sommets à l'extérieur du noyau avec témoin

Algorithme $\mathcal{A}_2$

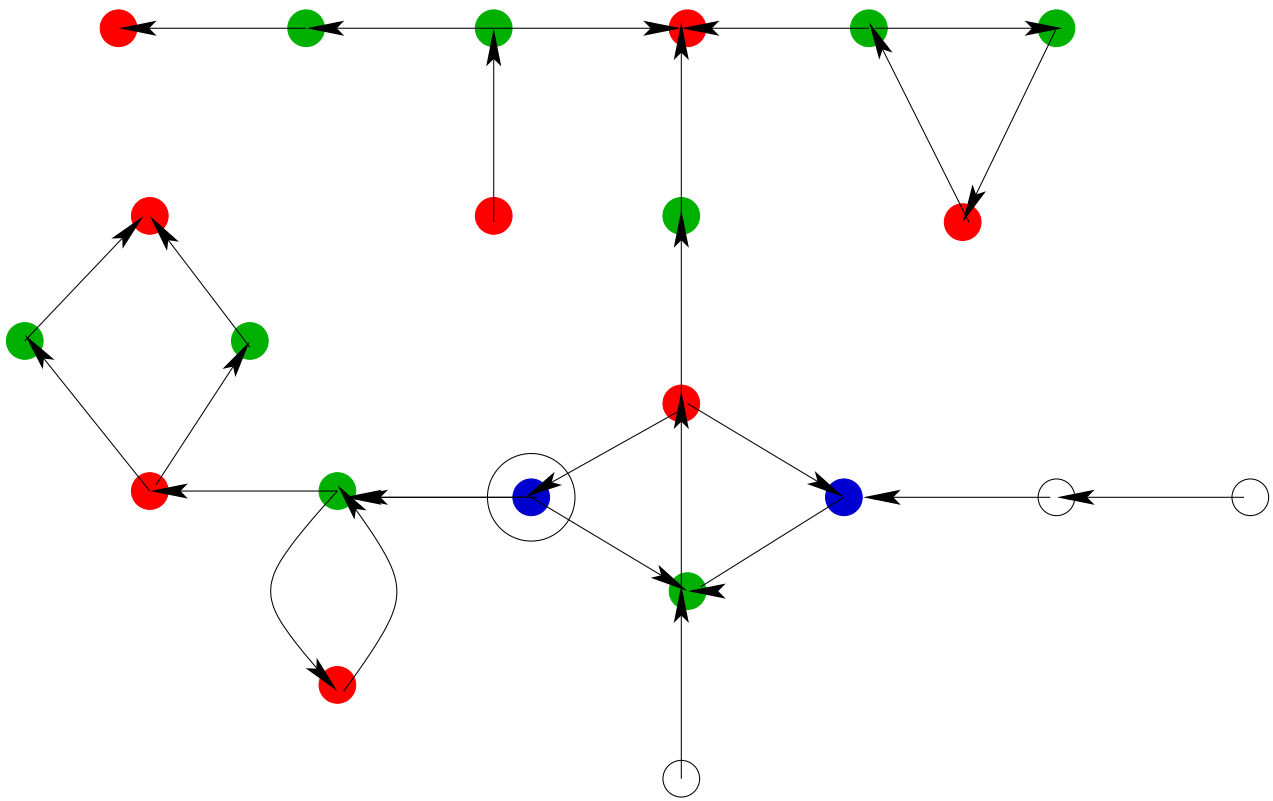
Tant que BLANC est non vide
  prendre $a \in$ BLANC
  énumérer les noyaux contenant a
  énumérer les noyaux ne contenant pas a

Nous obtenons un noyau lorsque tous les sommets sont coloriés en rouge et vert.

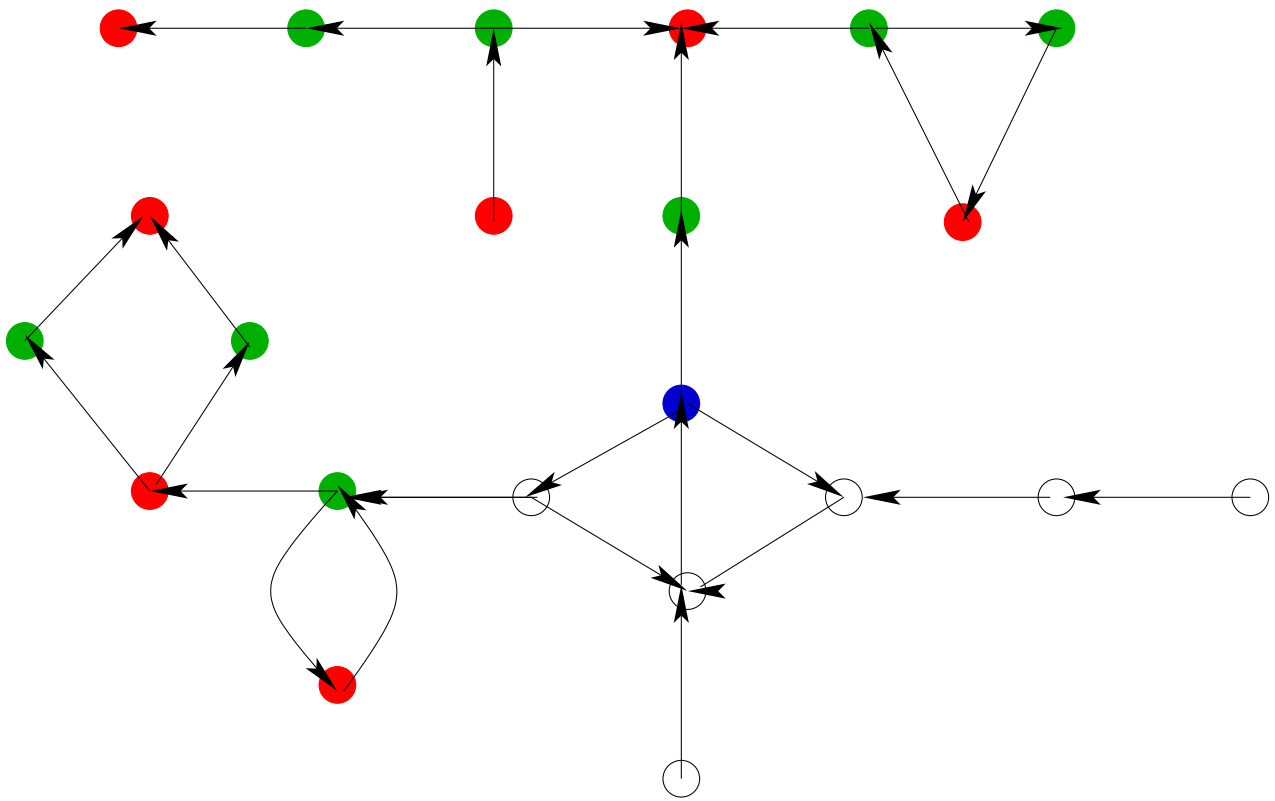
Algorithme $\mathcal{A}$ – Étape 2



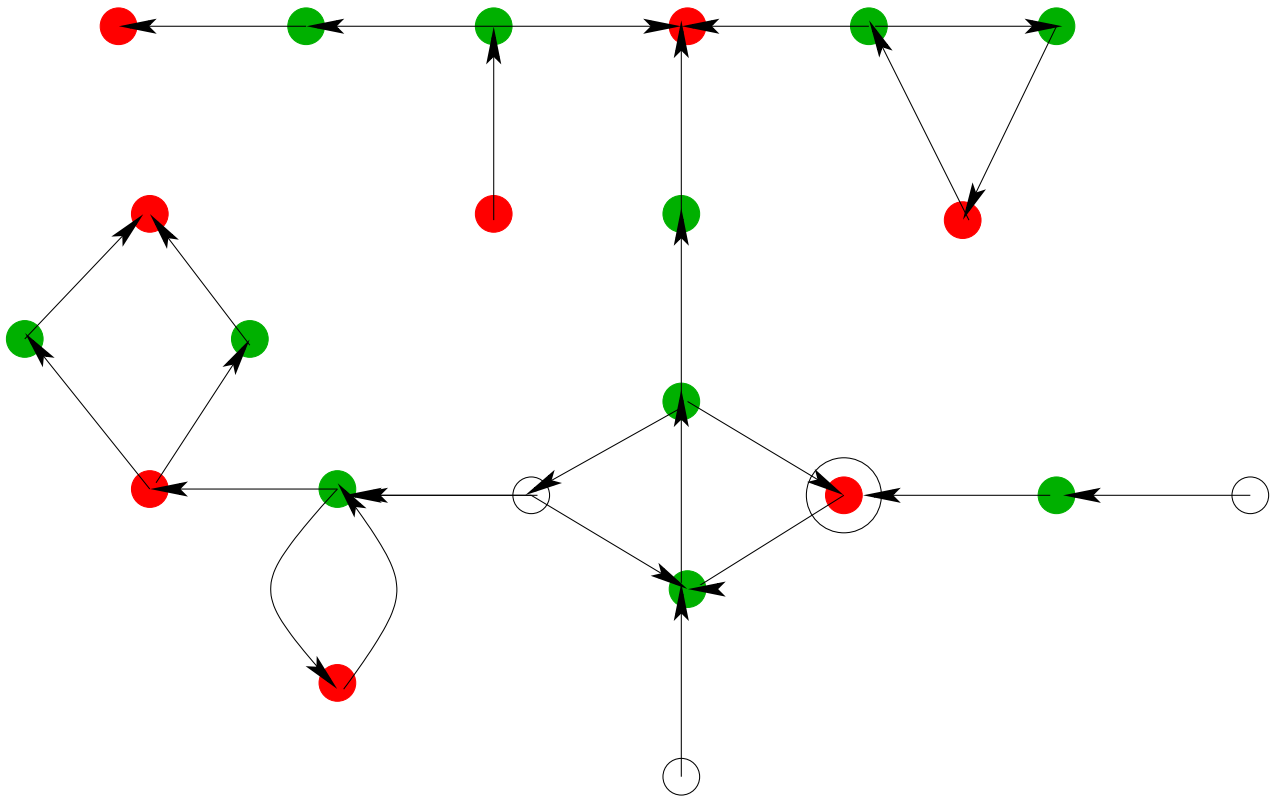
Algorithme $\mathcal{A}$ – Étape 2



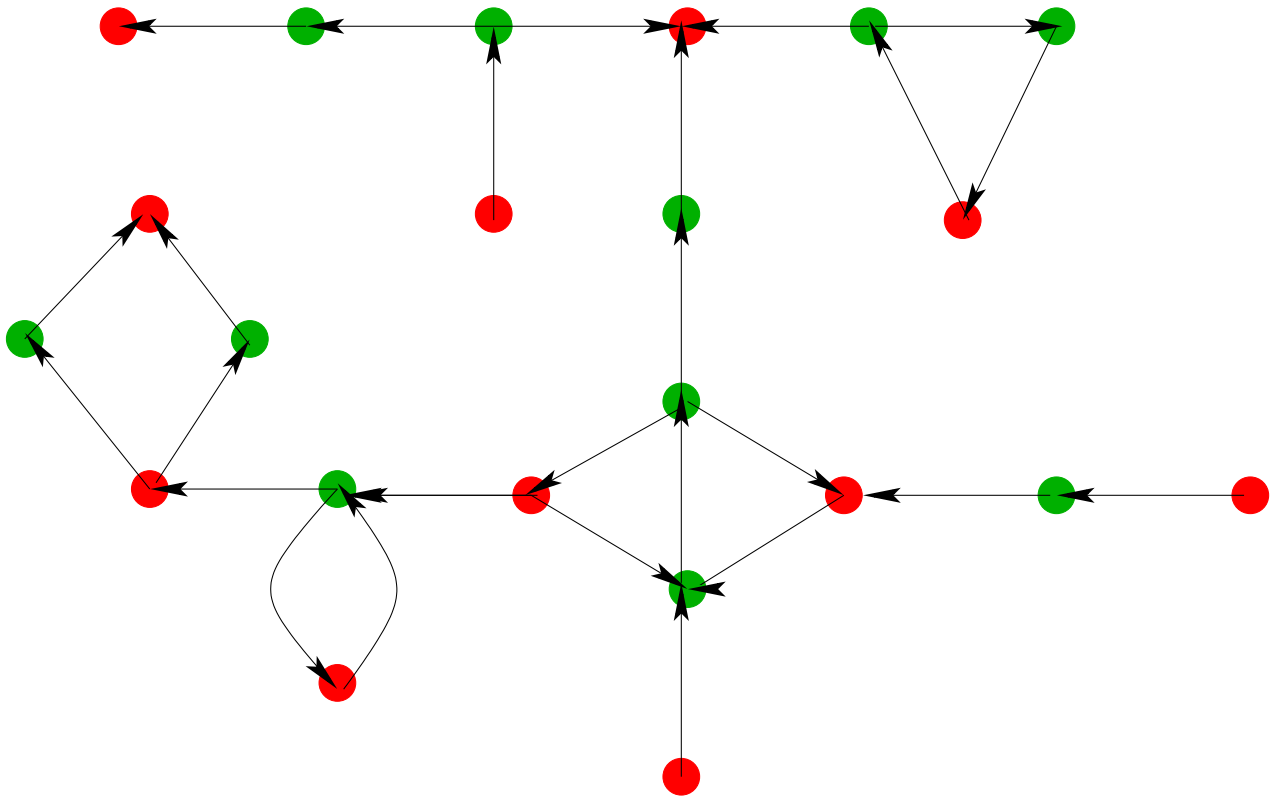
Algorithme $\mathcal{A}$ – Étape 2



Algorithme $\mathcal{A}$ – Étape 2



Algorithme $\mathcal{A}$ – Étape 2



Plan de l'exposé

Calculs préliminaires

$q = 1 - p$

Probabilités élémentaires

$S(q, r)$ (resp. $D(q, n, r)$ et $N(q, n, r)$) probabilité que U de taille r soit stable (resp. dominant, un noyau).

$$\begin{aligned} S(q, r) &= q^{r(r-1)} \\ D(q, n, r) &= (1 - q^r)^{n-r} \\ N(q, n, r) &= S(q, n, r)\, D(q, n, r). \end{aligned}$$

Espérance

Le nombre moyen de noyaux de taille r vaut donc

$$E[X_r^{Noyau}] = \binom{n}{r} q^{r(r-1)} (1 - q^r)^{n-r}.$$

Cas dense, p constant

Théorème Tomescu, Fernandez de la Vega, 1990

Notons $q = 1 - p$. Il existe un intervalle $\mathcal{I}(p, n)$ autour de $\beta(n) = \log_{1/q} n - \log_{1/q}(\log_{1/q} n)$ tel que les noyaux de $D_n \in \mathcal{D}(n, p)$ satisfont

1. a. p. s. tous les noyaux ont une taille dans $r \in \mathcal{I}(p, n)$;
2. pour toute suite $r(n) \in \mathcal{I}(p, n)$, D_n possède a. p. s. un noyau de taille $r(n)$.

Remarques

- la taille de $\mathcal{I}(n, p)$ ne dépend que de p
- l'intervalle diminue lorsque p augmente

Cas dense, p constant

Idée de la preuve

On utilise les Méthodes du premier et du second moment
Pour r proche de β, soit c le réel tel que $n = e^c\ r\ q^{-r}$.

$$E[X_r^{Noyau}] \sim \frac{e^{rg(c)}}{\sqrt{2\pi r}} \qquad var(X_r^{Noyau}) \sim E[X_r^{Noyau}]$$

où $g(c) = -e^c + c + 1 - \ln q$.

$r \in \mathcal{I}(n,p)$ si et seulement si $g(c) > 0$.

Complexité des algorithmes

Algorithme $\mathcal{B}$

$\mathcal{B}$ est un algorithme de Monte Carlo d'erreur $\lambda < 1/E[X_r^{Noyau}]$.
Sa complexité est superpolynomial $(n^{\ln n})$.

Algorithme $\mathcal{A}$

On *prouve* que l"algorithme $\mathcal{A}$ est également superpolynomial en remplaçant l'espérance de la complexité par la complexité de l'espérance.

Pour une preuve rigoureuse, il faut utiliser le modèle $\mathcal{D}(n, M)$.

Plan de l'exposé

Cas éparse, $p = c/n$

Calcul de l'espérance

Rappel

$$E[X_r^{Noyau}] = \binom{n}{r} S(r,q)\ D(n,r,q)$$

Pour $p = c/n$ et $r = \mu n$, μ densité de r

$$\begin{array}{rcccc} \binom{n}{\mu n}/n & \sim & \beta(c,\mu) & = & -\mu\log\mu - (1-\mu)\log(1-\mu), \\ S(\mu n, 1-c/n)/n & \sim & \iota(c,\mu) & = & -c\mu^2, \\ D(n,\mu n, 1-c/n)/n & \sim & \delta(c,\mu) & = & (1-\mu)(1-e^{-c\mu}), \end{array}$$

Une seule densité possible

Théorème 1

Tous les noyaux de D_n sont a. p. s. tous de taille

$$r = \mu_c n + o(n),$$

où $c = -\frac{\log \mu_c}{\mu_c}$.

Idée de la preuve

On montre que $E[X_r^{Noyau}] = e^{\varphi(c,\mu)n+o(n)}$, où

$$\varphi(c,\mu) = -\mu \log \mu - (1-\mu)\log(1-\mu) - c\mu^2 + (1-\mu)\log(1-e^{-c\mu}).$$

$$\begin{cases} \varphi(c,\mu) < 0 & \text{pour } \mu \neq \mu_c \\ \varphi(c,\mu) = 0 & \text{sinon} \end{cases}$$

Aucune taille probable

Théorème 2

Pour toute suite $r(n)$, D_n n'a a. p. s. pas de noyau de taille r.

Idée de la preuve

Soient $\mu = \mu_c$ et $r = \mu n + o(n)$
On utilise le résultat suivant

$$\ln E[X_r^{Noyau}] = -\frac{x^2}{n}\frac{(1-\log\mu)^2}{2\mu(1-\mu)} - \frac{1}{2}\log n - \log\left(\mu\sqrt{2\pi\mu(1-\mu)}\right) + \varepsilon(n,x),$$

où $\varepsilon(n,x) \in O(\frac{1}{n}) + O(\frac{x}{n}) + O(\frac{x^3}{n^2})$.

Somme des espérances

Théorème 3

L'espérance du nombre de noyaux de densité μ_c est supérieure à 1.

Idée de la preuve

Soient $\mu = \mu_c$, $f = o(n^{\frac{2}{3}})$ et $|x| \leq f$.
On considère l'espérance du nombre de noyaux de taille $r = \mu n + x$.

$$E[X_r^K] = \frac{e^{-\eta \frac{x^2}{n}}}{\zeta\sqrt{n}}(1 + o(1)), \eta = \frac{(1 - \log \mu)^2}{2\mu(1-\mu)}, \zeta = \mu\sqrt{2\pi\mu(1-\mu)}.$$

On somme les espérances sur $f \in \omega(\sqrt{n})$,

$$\lim_{n\to\infty} \sum_{|x|\leq|f|} E[X_r^K] = \frac{1}{\zeta}\int_{-\infty}^{\infty} e^{-\eta y^2} dy = \frac{\sqrt{\pi}}{\zeta\sqrt{\eta}} = \frac{\frac{1}{\mu}}{1 + \log\frac{1}{\mu}} > 1,$$

Complexité des algorithmes

Algorithme $\mathcal{B}$

$\mathcal{B}$ est un algorithme de Monte Carlo d'erreur $\lambda = e^{-\epsilon n}$ sur les ensembles de densité μ, avec $|\mu - \mu_c| < \delta$.
Sa complexité est exponentielle.

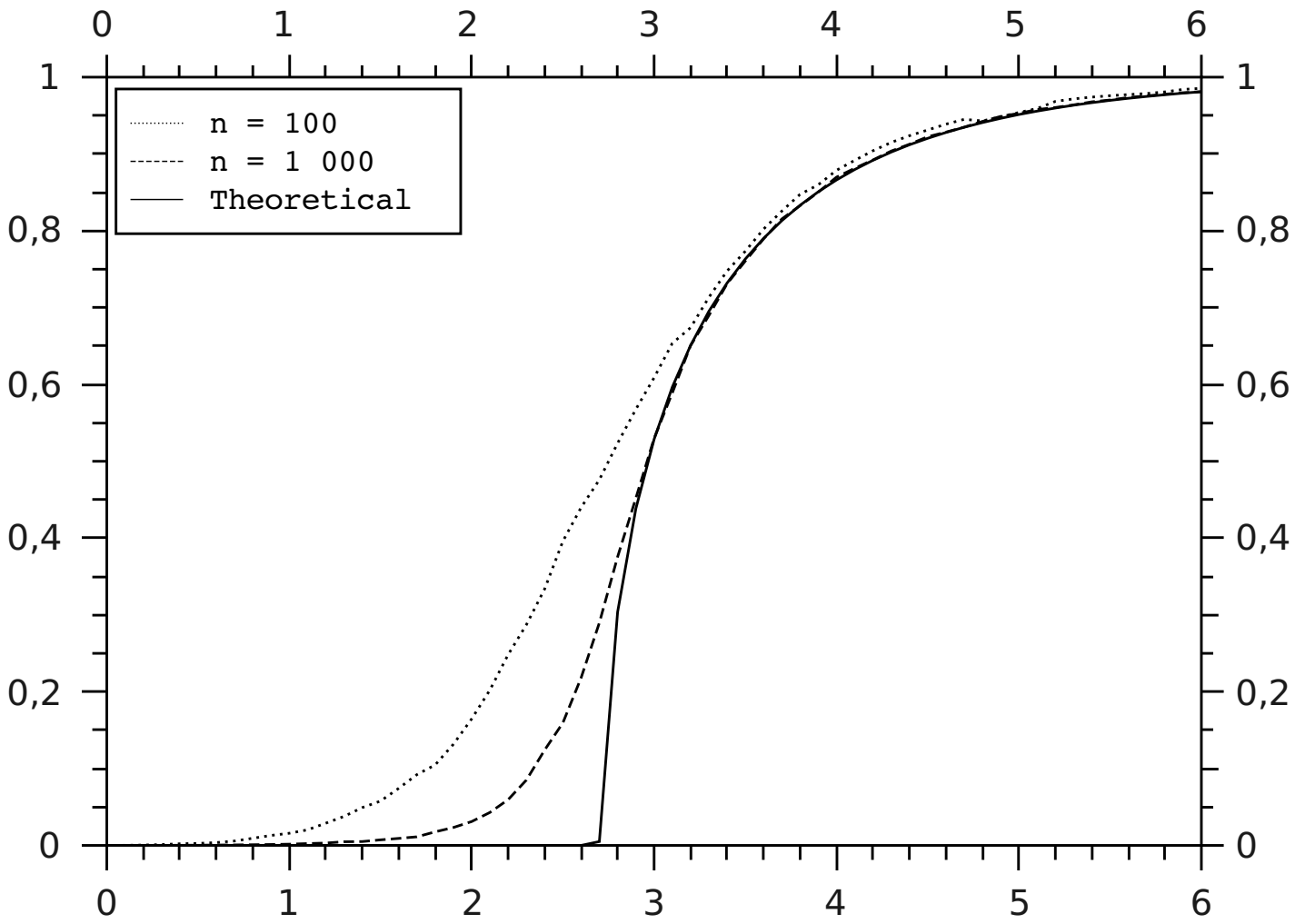
Algorithme $\mathcal{A}$

On *prouve* que l'algorithme $\mathcal{A}_2$ est exponentiel en la taille de BLANC, l'ensemble des sommets non coloriés par $\mathcal{A}_1$.

Complexité de $\mathcal{A}$

Dépend de la taille de BLANC.

Densité de BLANC après $\mathcal{A}_1$



Transition de phase en $c = 1$

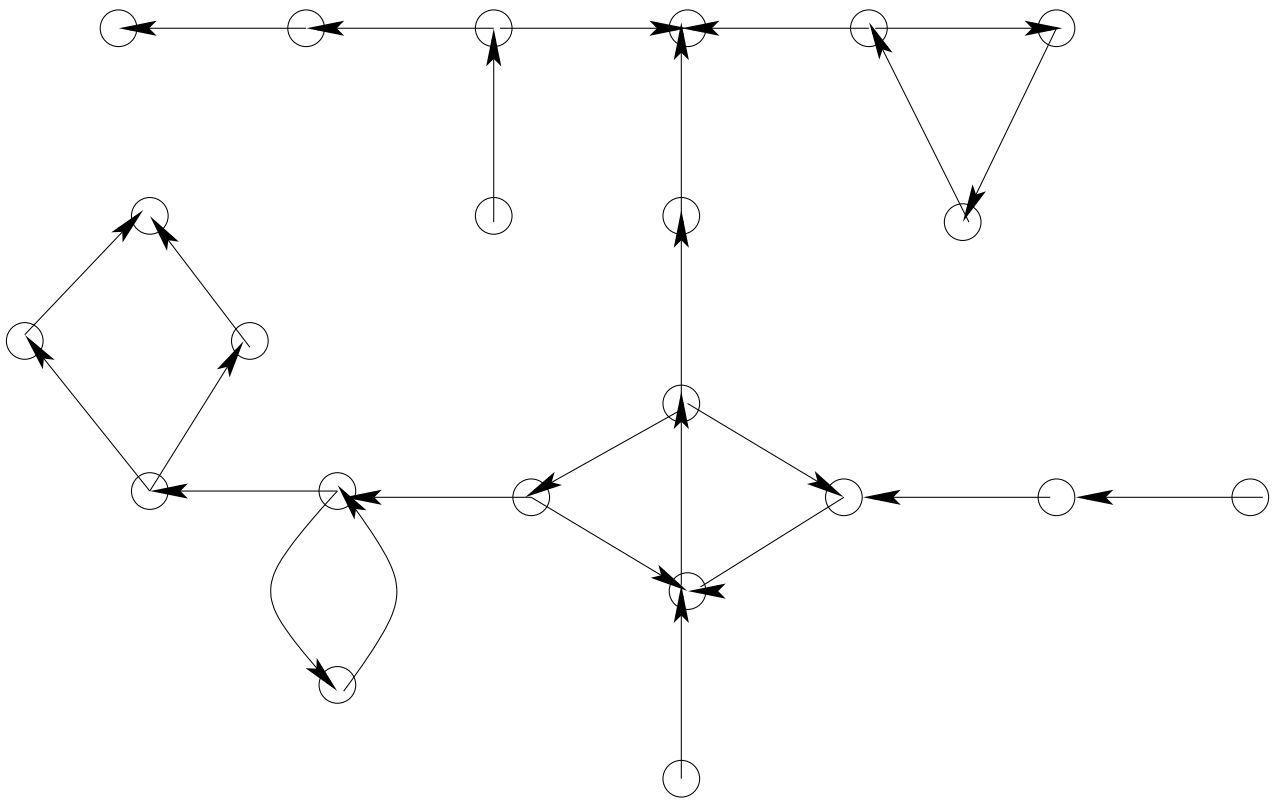
Expérimentations

On observe expérimentalement que tous les sommets sont coloriés avec l'algorithme $\mathcal{A}_1$, pour $c < 1$.

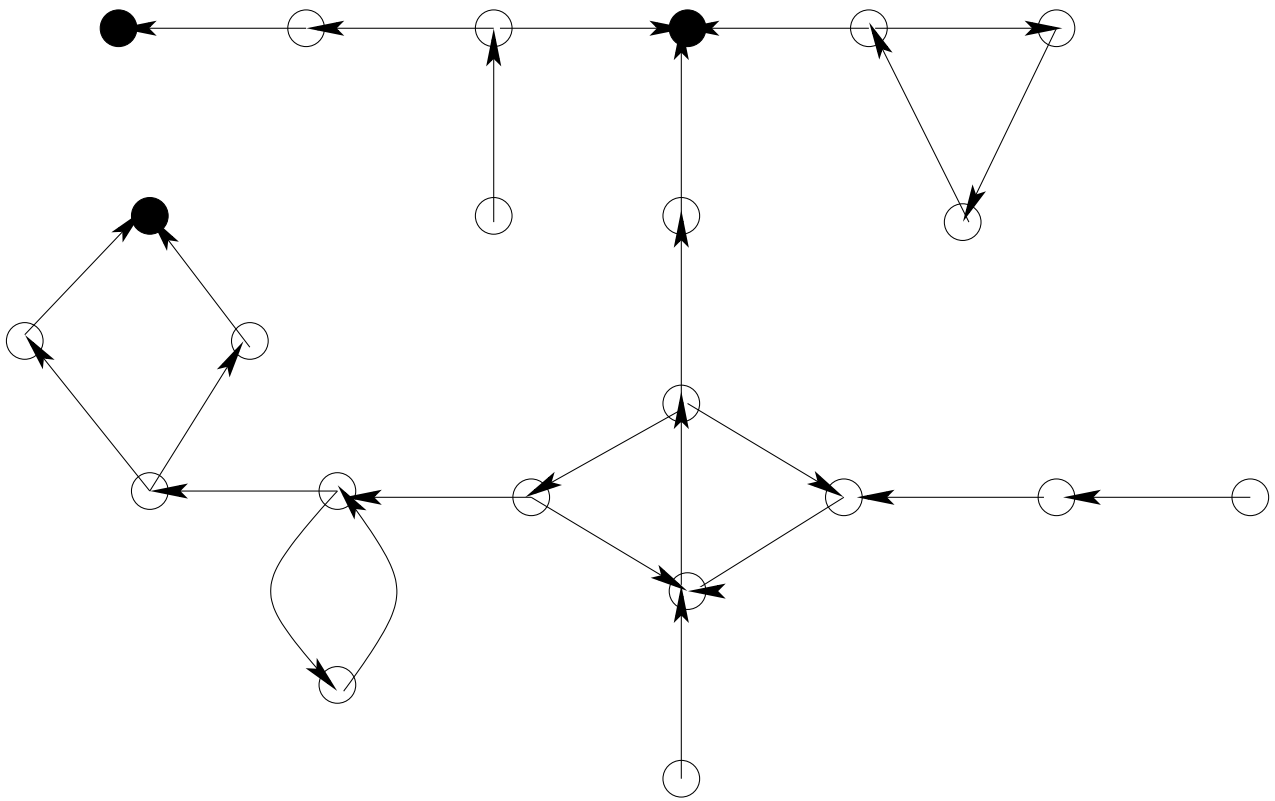
Coloriage en blanc et noir

On propose un coloriage plus grossier que celui de l'algorithme $\mathcal{A}_1$:
On colorie en noir tous les puits et tous les sommets allant vers un puits et n'appartenant pas à un circuit.

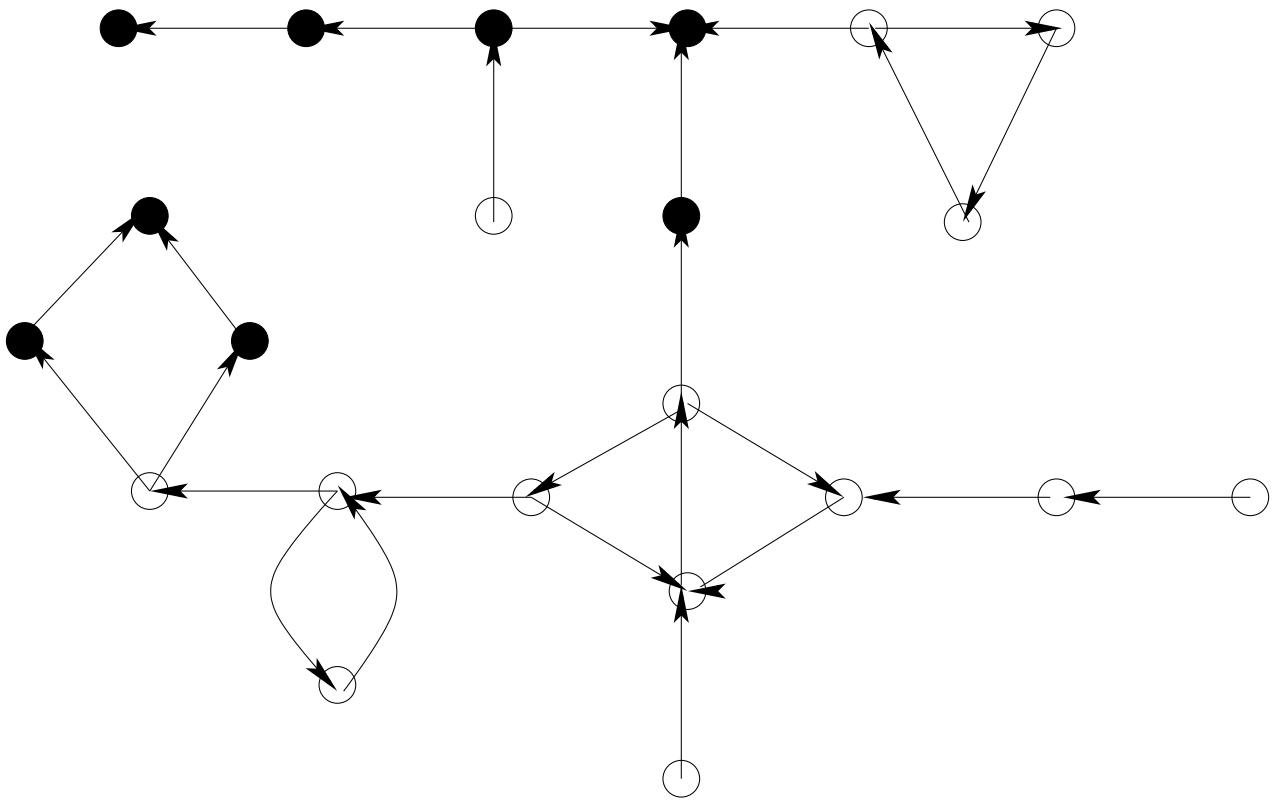
Algorithme $\mathcal{A}$ – Étape 1



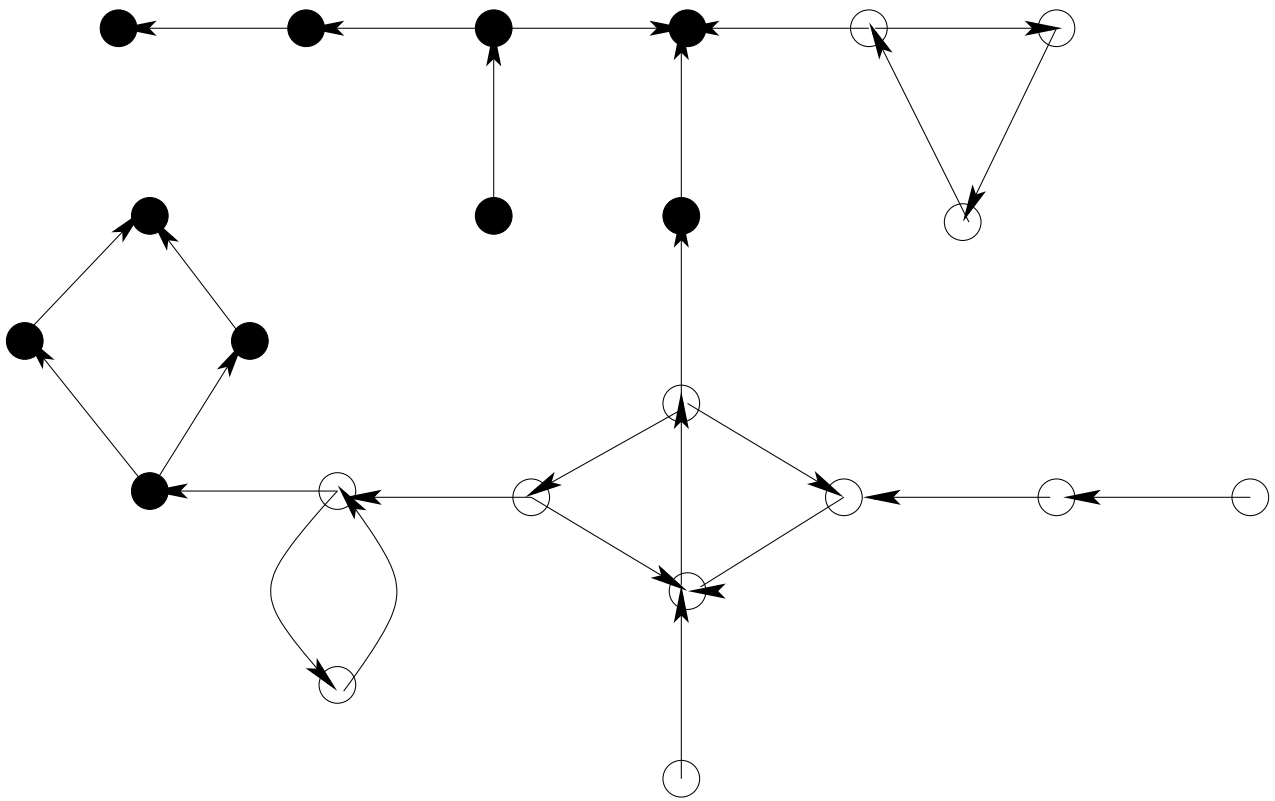
Algorithme $\mathcal{A}$ – Étape 1



Algorithme $\mathcal{A}$ – Étape 1



Algorithme $\mathcal{A}$ – Étape 1



Transition de phase en $c = 1$

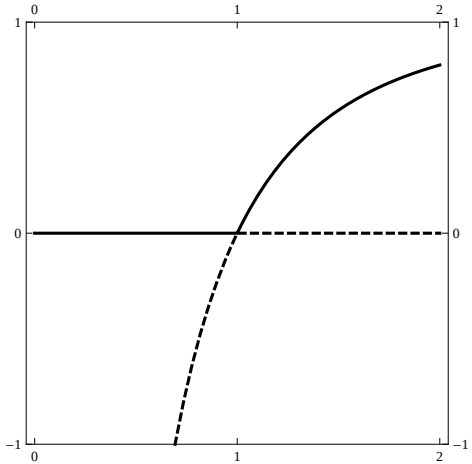
Coloriage en blanc et noir

w_i densité de BLANC à l'étape i, $|\text{BLANC}| \sim w_i n$.

Système dynamique $\begin{cases} w_0 &= 1 \\ w_{i+1} &= \varphi(w_i) \end{cases}$ où $\varphi(z) = 1 - e^{-cz}$.

Transition de phase

Soit $w = \lim_i \varphi^i(1)$. $\quad w = 0 \quad \text{si} \quad c < 1.$ $\quad w > 0 \quad \text{si} \quad c > 1.$



Transition de phase en $c = e$

Itérations de l'algorithme $\mathcal{A}_1$

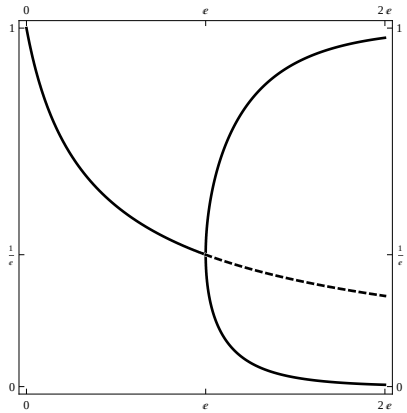
w_i et r_i densité de BLANC et ROUGE à l'itération i.

Système dynamique $\begin{cases} r_{i+1} &= \psi(1-g_i) \\ 1-g_i &= \psi(r_i) \end{cases}$ où $\psi(z) = e^{-cz}$

Transition de phase

$w = \lim_i \left(\psi^{2i+1}(0) - \psi^{2i}(0)\right)$.

$w = 0$ si $c < e$.
$w > 0$ si $c > e$.



Perspectives

Approfondir l'étude du cas $p = \frac{c}{n}$

- Etude de la variance.
- Mieux connaître la structure des noyaux pour garantir qu'il n'existe pas de (vraiment) meilleurs algorithmes.

Autres distributions $p = \frac{c \log n}{n}$

De nombreuses propriétés naturelles apparaissent avec cette distribution.

Merci de votre attention