

# Méthodes exactes et heuristiques pour la résolution du problème d'optimisation combinatoire : Burning Number

J. KOZIK

01-07-2025

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	The Burning Process . . . . .	2
1.2	The Burning Number Problem . . . . .	2
1.3	Burning Number and Process : Exemple . . . . .	3
1.3.1	Résultat . . . . .	5
<b>2</b>	<b>Travail effectué</b>	<b>5</b>
2.1	Objectif . . . . .	5
2.2	Modélisation Naive . . . . .	6
2.2.1	Etat/Noeud . . . . .	6
2.2.2	Résultats sur les instances . . . . .	7
2.3	Modélisation avec des balles . . . . .	8
2.3.1	Résultats sur les instances . . . . .	9
<b>3</b>	<b>Comparaison des modèles</b>	<b>9</b>
<b>4</b>	<b>Bibliographie</b>	<b>10</b>

## 1 Introduction

La propagation de l'influence sociale est un sujet majeur dans l'analyse des réseaux sociaux.

Une étude récente sur la contagion émotionnelle sur Facebook a mis en évidence que la structure du réseau sous-jacent joue un rôle essentiel, tandis que les interactions en personne ne sont pas déterminantes. Ainsi, les utilisateurs du réseau transmettent cette contagion à leurs amis et à leurs proches, ce qui permet au phénomène de continuer à se diffuser au fil du temps.

La question se pose alors : si l'objectif était de minimiser le nombre d'étapes nécessaires pour contaminer l'ensemble du réseau, quels utilisateurs devrait-on cibler en priorité, et dans quel ordre ?

En tant que modèle simplifié de ce problème, on peut utiliser un procédé sur les graphes appelé le *burning*.

### 1.1 The Burning Process

Soit un graphe  $G = (B, NB, E)$ .

Dans le procédé du *burning*, chaque sommet appartient à l'un des deux ensembles : *burned*( $B$ ) ou *unburned*( $NB$ ).

À chaque itération, on choisit un sommet à brûler.

Lorsqu'un sommet  $n$  est brûlé à l'itération  $i$ , alors à l'itération  $i + 1$ , tous les sommets *unburned* de son voisinage sont brûlés.

Le procédé se termine une fois que tous les sommets sont brûlés (ie. l'ensemble  $NB$  est vide).

### 1.2 The Burning Number Problem

On appelle le *burning number* d'un graphe  $G$ , écrit  $b(G)$ , le nombre minimum d'itération nécessaire pour terminer le *burning process*.

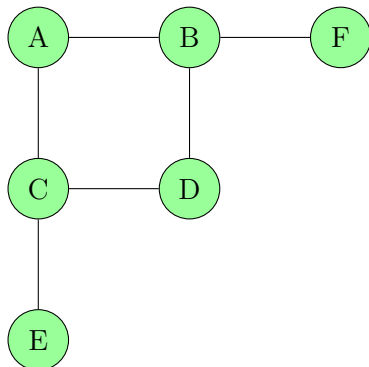
Pour certaines classes de graphes, le burning number est connu, cependant de manière générale, il existe une conjecture postulant que pour tout graphe à  $n$  sommets,  $b(G) \leq \lceil \sqrt{n} \rceil$ . Pour le travail effectué lors du stage, les résultats importants sont que les graphes cycles et chemins testés atteignent la borne de la conjecture. Le problème du burning number est classé comme étant NP-complet.

### 1.3 Burning Number and Process : Exemple

#### Initialisation

Soit  $G$  :

- $NB : \{A, B, C, D, E, F\}$
- $B : \{\}$
- $E : \{AB, AC, BD, CD, CE, BF\}$

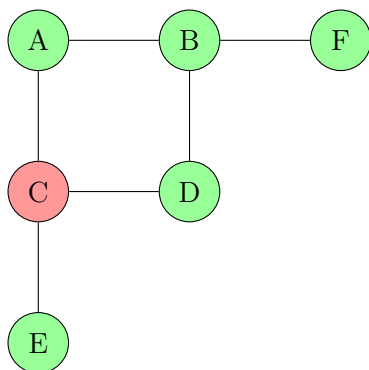


#### Etape 1A : Propagation

Aucun sommet n'a été brûlé à l'étape 0, donc le graphe reste inchangé.

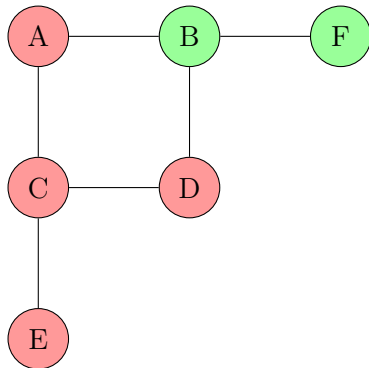
#### Etape 1B : Choix d'un sommet : C

- $NB : \{A, B, D, E, F\}$
- $B : \{C\}$
- $E : \{AB, AC, BD, CD, CE, BF\}$

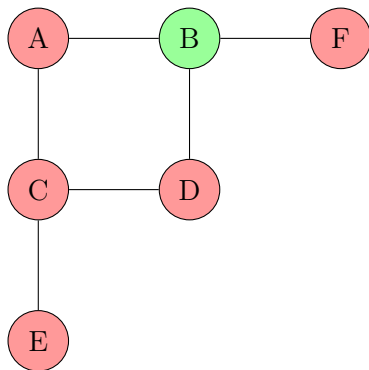


**Etape 2A : Propagation**

- ▶  $NB : \{B, F\}$
- ▶  $B : \{A, C, D, E\}$
- ▶  $E : \{AB, AC, BD, CD, CE, BF\}$

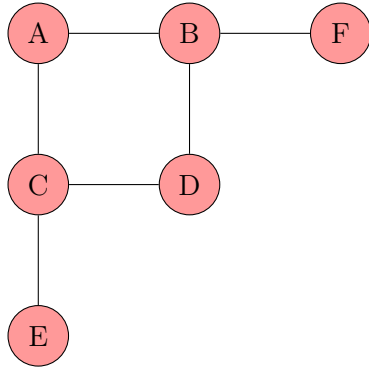
**Etape 2B : Choix d'un sommet : F**

- ▶  $NB : \{B\}$
- ▶  $B : \{A, C, D, E, F\}$
- ▶  $E : \{AB, AC, BD, CD, CE, BF\}$



**Etape 3A : Propagation**

- ▶  $NB : \{\}$
- ▶  $B : \{A, B, C, D, E, F\}$
- ▶  $E : \{AB, AC, BD, CD, CE, BF\}$

**Etape 3B : choix d'un sommet**

Puisque l'ensemble des sommets  $NB$  est vide, le choix pour le troisième sommet est arbitraire.

**1.3.1 Résultat**

Le *burning process* se termine après un total de 3 itération. On en conclut que  $b(G) = 3$ , et la séquence qui permet d'arriver à ce résultat est  $\{C, F, ?\}$ .

**2 Travail effectué****2.1 Objectif**

L'objectif du stage était de s'appuyer sur le paradigme de la recherche arborescente et la programmation dynamique pour modéliser et implémenter et évaluer des méthodes permettant de retrouver des bonnes solutions pour le problème du Burning Number.

## 2.2 Modélisation Naive

Dans un premier temps, l'objectif du stage était de créer une modélisation naïve du problème afin d'obtenir une solution rapide, même si celle-ci pouvait être moins précise.

### 2.2.1 Etat/Noeud

A chaque itération du *burning process* un noeud  $v$  décrit :

- ▶ un graphe connexe  $G = (B, NB, E)$
- ▶  $N$  : l'ensemble des sommets brûlés.
- ▶  $NB$  : l'ensemble des sommets non-brûlés.
- ▶  $E$  : l'ensemble des arêtes.
- ▶  $n$  : le numéro de la séquence.
- ▶  $B_v$  : la liste de sommets  $v$  brûlés par la balle de taille  $B$ .
- ▶  $C$  : la liste de sommet au centre de chaque balle.
- ▶ *score* : un score associé à l'état.

### Calcul du score pour la version naive

Afin de déterminer le *burning number* minimal, on associe à chaque nœud  $v \in V$  un score  $s(v)$  défini par :

$$s(v) = \sum_{i=0}^{|B_v|} \frac{|B_v[i]|}{i^2} \quad (1)$$

### 2.2.2 Résultats sur les instances

TABLE 1 – Résultats obtenus avec la modélisation naïve. Où,  $V$  représente le nombre de sommets,  $E$  le nombre d'arêtes,  $U$  la borne supérieure,  $b(G)$  la valeur optimale du *burning number*,  $Gap$  la différence entre  $U$  et  $b(G)$ , et  $R$  le résultat obtenu avec le modèle naïf.

Instances	$V$	$E$	$U$	$b(G)$	Gap	$R$
karate	34	78	3	2	0%	3
chesapeake	39	170	2	2	0%	2
dolphins	62	159	4	3	29%	3
rt-retweet	96	117	5	4	32%	5
polbooks	105	441	3	3	5%	3
adjnoun	112	425	3	3	0%	3
ia-infect-hyper	113	2196	2	2	7%	3
C125-9	125	6963	5	2	46%	4
ia-enron-only	143	623	4	3	11%	4
c-fat200-1	200	1534	6	6	3%	7
c-fat200-2	200	3235	4	4	0%	4
c-fat200-5	200	8473	2	2	0%	2
sphere	258	1026	7	6	40%	7
DD244	291	822	8	6	58%	7
ca-netscience	379	914	7	5	57%	7
infect-dublin	410	2765	5	4	55%	5
c-fat500-1	500	4459	10	8	35%	10
c-fat500-2	500	9139	6	6	0%	6
c-fat500-5	500	23191	2	4	15%	4
bio-diseaseome	516	1188	7	6	42%	7
web-polblogs	643	2280	5	4	0%	5
DD687	725	2600	9	6	65%	8

## 2.3 Modélisation avec des balles

Dans un second temps, l'objectif du stage était de concevoir une modélisation basée sur l'utilisation de balles afin d'améliorer la précision de la solution, quitte à augmenter le temps de calcul nécessaire pour l'obtenir.

La modélisation du noeud ne change pas, seulement le calcul du score.

### Calcul du score pour la modélisation par balles

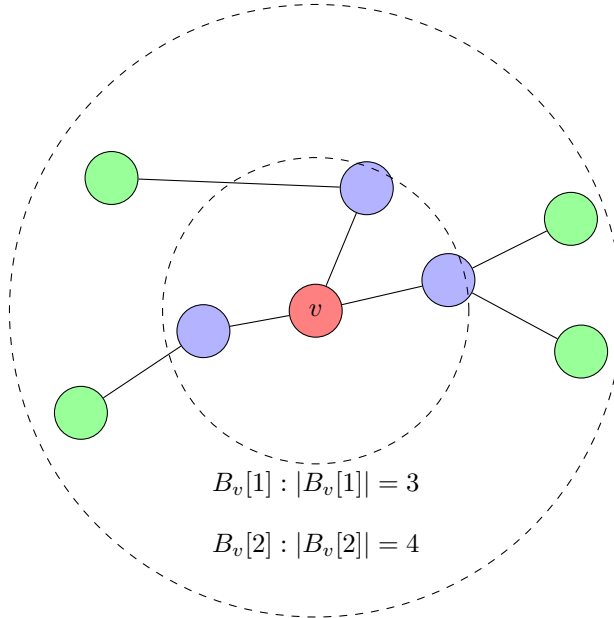
Nous définissons, pour chaque sommet  $v$ , une famille de balles  $B_v[i]$  correspondant à l'ensemble des sommets situés à distance exactement  $i$  de  $v$ . Le score associé à  $v$  est alors donné par :

$$s(v) = \sum_{i=0}^{|B_v|-1} \text{score\_sommet}(v, i) - \text{diam}(N_B)^2, \quad (2)$$

où  $|B_v[i]|$  désigne la cardinalité de la  $i$ -ième balle autour de  $v$ , et  $\text{diam}(N_B)$  le diamètre du sous-graphe considéré.

La contribution locale est définie par :

$$\text{score\_sommet}(v, i) = \begin{cases} |B_v[0]|, & \text{si } i = 0, \\ \frac{|B_v[i]|}{i^2}, & \text{si } i \geq 1. \end{cases} \quad (3)$$





### 2.3.1 Résultats sur les instances

TABLE 2 – Résultats obtenus avec la modélisation par balles. Où,  $V$  représente le nombre de sommets,  $E$  le nombre d'arêtes,  $U$  la borne supérieure,  $b(G)$  la valeur optimale du *burning number*,  $Gap$  la différence entre  $U$  et  $b(G)$ , et  $R$  le résultat obtenu avec le modèle par balles.

Instances	$V$	$E$	$U$	$b(G)$	Gap	$R$
karate	34	78	3	2	0%	2
chesapeake	39	170	2	2	0%	2
dolphins	62	159	4	3	29%	3
rt-retweet	96	117	5	4	32%	4
polbooks	105	441	3	3	5%	3
adjnoun	112	425	3	3	0%	3
ia-infect-hyper	113	2196	2	2	7%	2
C125-9	125	6963	5	2	46%	3
ia-enron-only	143	623	4	3	11%	3
c-fat200-1	200	1534	6	6	3%	6
c-fat200-2	200	3235	4	4	0%	4
c-fat200-5	200	8473	2	2	0%	2
sphere	258	1026	7	6	40%	6
DD244	291	822	8	6	58%	7
ca-netscience	379	914	7	5	57%	5
infect-dublin	410	2765	5	4	55%	4
c-fat500-1	500	4459	10	8	35%	-
c-fat500-2	500	9139	6	6	0%	-
c-fat500-5	500	23191	2	4	15%	-
bio-diseaseome	516	1188	7	6	42%	-
web-polblogs	643	2280	5	4	0%	-
DD687	725	2600	9	6	65%	-

## 3 Comparaison des modèles

Les deux modèles proposés ne résolvent le problème qu'approximativement et ne garantissent pas l'optimalité pour toutes les instances. Toutefois, le second modèle montre une nette supériorité. Le modèle naïf génère des solutions plus faibles, atteint rapidement les limites imposées par les instances, et peut même les dépasser, indiquant une moins bonne maîtrise de l'espace de recherche.

## 4 Bibliographie

1. Anthony Bonato, Jeannette Janssen, Elham Roshanbin - *How to burn a graph* - 2015
2. Peter Norvig, Stuart Russell - *Artificial Intelligence : A Modern Approach* - Third Edition - 2016
3. Stéphane Bessy, Anthony Bonato, Jeannette Janssen, Dieter Rautenbach, Elham Roshanbin - *Bounds on the burning number* - 2016

## Remerciement

Je tiens à exprimer ma profonde gratitude à Rafael Colares Borges, dont l'accompagnement, les conseils et la disponibilité ont grandement contribué à mon apprentissage durant ce stage.

Je remercie également ma responsable de licence, Fatiha Bendali-Mailfert, pour son suivi, ses retours et son soutien tout au long de cette période.

Enfin, j'adresse mes remerciements aux trois étudiants qui ont partagé ce stage avec moi. Leur aide, leur bonne humeur et nos échanges quotidiens ont rendu cette expérience à la fois enrichissante et agréable.