



N° Réf :.....

Université de Mila

Faculté des Mathématiques et Informatique

Département de Mathématiques

**Mémoire préparé En vue de l'obtention du diplôme de Master
En: Mathématiques**

**Spécialité:
Modélisation Mathématiques et Techniques de Décision**

Secure broadcast domination dans les graphes

Préparé par :

- Laib Fayza
- Makouf Sarra

Soutenue le .../06/2026

devant le jury

Azi Mourad	MCB	U. De, Mila	Président
Boufelgha Ibrahim	MCB	U. De, Mila	Rapporteur
Guettiche Mourad	MCB	U. De, Mila	Co-Rapporteur
Boudebane Zineb	MCB	U. De, Mila	Examineur

Année universitaire : 2025/2026

Remerciements

Avant tout, nous rendons grâce à Dieu Tout-Puissant, source inépuisable de force et de lumière, qui nous a guidés pas à pas et nous a accordé la volonté et la patience nécessaires pour accomplir ce travail.

Nous tenons à exprimer notre profonde gratitude et nos remerciements les plus sincères à nos estimés encadreurs, Monsieur "**Boufelgha Ibrahim**" et Monsieur "**Guet-tiche Mourad**". Nous les remercions chaleureusement pour leur aide précieuse, leurs conseils judicieux, leur grande patience et le soutien constant qu'ils nous ont prodigué tout au long de la réalisation de ce projet. Ce travail n'aurait pas pu voir le jour sans leur encadrement exemplaire.

Nous adressons également nos vifs remerciements au **Président du jury** ainsi qu'aux honorables **membres du jury** qui ont accepté d'évaluer notre travail, de l'enrichir par leurs remarques et de siéger à notre soutenance. C'est un grand honneur et une immense fierté pour nous de présenter notre mémoire devant eux.

Enfin, nous tenons à remercier chaleureusement toutes les personnes qui ont contribué de près ou de loin à l'aboutissement de ce mémoire, que ce soit par un mot d'encouragement, un geste ou une simple pensée.

Fayza et Sara

Dédicace

*Louange à Dieu qui a éclairé mon chemin par la lumière du savoir, et
Qui m'a accordé la force, la patience et la persévérance pour mener à
Bien ce humble travail tout au long de mon cursus universitaire*

A mes chers parents, mon guide et mon soutien dans cette vie :

*Mon tender père "**Messaoud**" et ma douce mère "**Malika**"*

*Que Dieu vous préserve, vous accorde santé et longue vie, et vous garde
Comme une couronne de fierté sur ma Tête*

*A ces cœurs généreux qui ont toujours été pour moi une source
d'encouragement, de joie et d'énergie positive*

*A mon cher frère **Yassine**,*

Mon précieux soutien, et à mes douces Sœurs :

Asma Fatima Ilham Rania

*A la joie de mon cœur et au bonheur de notre famille Mes chers neveux
et nieces, les amours de leur tante A nos tendres rayons de soleil :*

Maram Malak Assil Alaa Amir Younes Moumen

*A celle qui a été ma meilleure compagne de route, avec qui j'ai partagé
les moments de rigueur et le stress A celle qui a partagé avec moi ce
voyage du savoir, les veilles et les efforts de chaque jour ... Ma chéré
collègue et amie **Sarra***

FAYZA

Dédicace

Louange à Dieu qui a éclairé mon chemin par la lumière du savoir, et qui m'a accordé la force, la patience et la persévérance pour mener à bien cet humble travail tout au long de mon cursus universitaire .

A mes chers parents, mon guide et mon soutien dans cette vie :

Mon tendre père et ma douce mère

Que Dieu vous préserve, vous accorde santé et longue vie, et vous garde

Comme une couronne de fierté sur ma tête

A ces cœurs généreux qui ont toujours été pour moi une source d'encouragement, de joie et d'énergie positive

A mon cher frère, mon précieux soutien, et à ma douce sœur

*A celle qui a été ma meilleure compagne de route, avec qui j'ai partagé les moments de rigueur et le stress et partagé avec moi ce voyage du savoir, les veilles et les efforts de chaque jour ... Ma chère collègue et amie **Fayza***

Sarra

Table des matières

1	Concepts fondamentaux	13
1.1	Notions et terminologie	14
1.2	Paramètres des graphes	19
1.3	Quelques classes de graphes	21
2	Domination et broadcasts domination dans les graphes	27
2.1	Domination dans les graphes	28
2.2	Les variantes de la domination	30
2.3	Broadcast domination	35
2.4	Les variantes de la broadcast domination	36
3	Broadcast domination sécurisée	38
3.1	Définitions et notations	39
3.2	Bornes sur γ_{bs}	39
3.3	Broadcast domination sécurisée sur des classes de graphes	41
4	Amélioration et sécurisation de l'ensemble broadcast dominant	44
4.1	L'heuristique proposée	45
4.2	Analyse de la complexité	56
4.3	Validation et analyse des performances	57
4.4	Évaluation de la fiabilité et de l'efficacité de l'approche	61

Table des figures

1.1	Graphe non orienté	14
1.2	Graphe orienté	15
1.3	Multigraphe	15
1.4	Graphe non connexe mais les composantes (C1) (C2) sont connexes . . .	16
1.5	Stable maximal et stable maximum	20
1.6	Coloration propre des sommets d'un graphe G	21
1.7	Graphe complet K_5	22
1.8	Cycle C_6	23
1.9	L'arbre contient une racine 1, des sommets internes 2 et 3, et 4,5,6,7,8 des feuilles.	24
1.10	G et son complément $\bar{G}$	25
1.11	Graphe biparti et son graphe biparti complet $K_{4,3}$	25
2.1	Un graphe G avec $\gamma(G) = 2$ et $\Gamma(G) = 3$	28
2.2	Ensemble dominant indépendant	30
2.3	Ensemble dominant double $S = \{v_1, v_2, v_3, v_5\}$, est ce que S est minimum $\gamma_{\times 2}(G) = 4$	31
2.4	Ensemble dominant total : $S = \{v_2, v_4, v_7\}$	32
2.5	Un graphe G avec $\gamma_R(G) = 4$	33
2.6	La chaine P_{10} avec $S = \{2, 4, 6, 8, 10\}$	34
2.7	Le graphe complet K_4 avec $S = \{v_2\}$	34
2.8	L'ensemble broadcast est $V_f^+ = \{x_2\}$ et $\gamma_b(G) = 2$	35
2.9	Un broadcast stable non efficace sur le graphe P_6	37
2.10	Un packing maximal qui n'est pas un broadcast packing maximal	37

3.1	Chaine P_{11} avec un broadcast sécurisé : $\gamma_{bs}(P_{11}) = 5$	42
4.1	Application de l'algorithme lokshtanov	47
4.2	Comparaison des paramètres γ_b et γ_{bs} pour les instances synthétiques.	62
4.3	Analyse du temps d'exécution par phase (Instances synthétiques)	62
4.4	Comparaison des paramètres γ_b et γ_{bs} pour les instances réelles.	63
4.5	Analyse du temps d'exécution par phase (Instances réelles)	64

Résumé

Le problème de la domination dans les graphes (Graph Domination) constitue l'un des axes de recherche les plus cruciaux en théorie des graphes, en raison de ses applications vitales dans l'optimisation de l'efficacité et de la sécurité des réseaux de communication sans fil. Ce travail de recherche est dédié à l'étude du modèle de la **domination par broadcast sécurisée (Secure Broadcast Domination)**. Ce concept, qui représente une extension avancée de la domination classique, vise à déterminer une distribution optimale de la puissance de broadcast entre les sommets afin de garantir une couverture complète du réseau, tout en préservant la sécurité des données face aux intrusions ou aux attaques externes.

Dans le cadre de ce mémoire, nous allons analyser les propriétés structurelles et mathématiques des graphes sous les contraintes de sécurité et de broadcast, et développer des algorithmes pour calculer le "nombre de domination par broadcast sécurisée" (γ_{bs}) pour plusieurs familles de graphes.

De plus, nous allons évaluer les performances de ces algorithmes ainsi que leur complexité computationnelle et leur extensibilité (Scalability) sur des réseaux dont la taille varie de 250 à 1000 sommets, en utilisant le langage de programmation **Python** et la bibliothèque spécialisée **NetworkX**. Les résultats obtenus démontrent l'efficacité du modèle proposé à instaurer un compromis idéal entre la minimisation de la consommation d'énergie de broadcast et le maintien d'un niveau maximal de protection et de couverture au sein du réseau.

Mots-clés : Domination dans les graphes, Broadcast Domination, Secure Broadcast Domination, Ensemble dominant sécurisé, Nombre de domination sécurisée (γ_{bs}), Sécurité des réseaux de communication, Simulation avec **NetworkX**.

Abstract

The graph domination problem is one of the most important research areas in graph theory due to its significant applications in improving the efficiency and security of wireless communication networks. This research work is devoted to the study of the **Secure Broadcast Domination** model. This concept, which represents an advanced extension of classical domination, aims to determine an optimal distribution of broadcast power among the vertices in order to ensure complete network coverage while maintaining data security against intrusions and external attacks.

Within the framework of this thesis, we analyzed the structural and mathematical properties of graphs under security and broadcast constraints and developed algorithms for computing the *secure broadcast domination number* (γ_{bs}) for several graph families.

Furthermore, we evaluated the performance, computational complexity, and scalability of these algorithms on networks ranging from 250 to 1000 vertices, using the **Python** programming language and the specialized **NetworkX** library. The obtained results demonstrate the effectiveness of the proposed model in achieving an optimal trade-off between minimizing broadcast energy consumption and maintaining a high level of security and coverage throughout the network.

Keywords : Graph Domination, Broadcast Domination, Secure Broadcast Domination, Secure Dominating Set, Secure Broadcast Domination Number (γ_{bs}), Communication Network Security, NetworkX Simulation.

ملخص

تُعدّ مسألة السيطرة في الرسوم البيانية (Graph Domination) أحد أهم مجالات البحث في نظرية الرسوم البيانية، وذلك بسبب تطبيقاتها الحيوية في تحسين كفاءة وأمن شبكات الاتصال اللاسلكية. يهدف هذا العمل البحثي إلى دراسة نموذج السيطرة بال بثّ الأمان (Secure Broadcast Domination) ويُعدّ هذا المفهوم امتداداً متقدماً لمفهوم السيطرة التقليدية، حيث يسعى إلى تحديد توزيع أمثل لقدرة البثّ بين الرؤوس، بما يضمن تغطية كاملة للشبكة مع الحفاظ في الوقت نفسه على أمن البيانات ضد الاختراقات والهجمات الخارجية.

في إطار هذه المذكرة، قمنا بدراسة الخصائص البنائية والرياضية للرسوم البيانية تحت قيود الأمان والبثّ، كما قمنا بتطوير خوارزميات لحساب عدد السيطرة بال بثّ الأمان (γ_{sb}) لعدة عائلات من الرسوم البيانية.

بالإضافة إلى ذلك، قمنا بتقييم أداء هذه الخوارزميات وتعقيدها الحسابي وقابليتها للتوسع (Scalability) على شبكات يتراوح حجمها بين 250 و 1000 رأس، باستخدام لغة البرمجة Python والمكتبة المتخصصة NetworkX. أظهرت النتائج أن النموذج المقترح يحقق توازناً فعالاً بين تقليل استهلاك طاقة البثّ والحفاظ على مستوى عالٍ من الحماية والتغطية داخل الشبكة.

الكلمات الأساسية: السيطرة في الرسوم البيانية، السيطرة بال بثّ (Broadcast Domination)، السيطرة بال بثّ الأمان (Secure Broadcast Domination)، مجموعة مسيطرة آمنة، عدد السيطرة الأمانة (γ_{sb})، أمن شبكات الاتصالات، المحاكاة باستخدام NetworkX.

Introduction

L'origine de l'optimisation remonte à la recherche de techniques rationnelles pour l'analyse et la résolution de problèmes complexes. À cet égard, la recherche opérationnelle, basée sur la modélisation mathématique, constitue le fondement essentiel visant à élaborer des décisions efficaces pour l'obtention des meilleurs résultats possibles. Dans ce contexte, la théorie des graphes est reconnue comme un puissant outil de modélisation et de résolution des problèmes combinatoires. C'est dans ce cadre que s'inscrit notre travail, dédié à l'étude d'une variante de la domination.

L'histoire de la théorie des graphes débute avec les travaux d'Euler au XVIII^{ème} siècle, et trouve son origine dans l'étude de certains problèmes, tels que celui des ponts de Königsberg. Les habitants de Königsberg se devaient de résoudre l'énigme de savoir s'il était possible, en partant d'un quartier quelconque de la ville, de traverser tous les ponts sans passer deux fois par le même, pour enfin revenir à leur point de départ [19].

Commencer par le concept de *broadcast domination* a été introduit comme une généralisation naturelle de la domination standard dans les graphes, offrant un modèle flexible où chaque sommet peut agir comme un broadcast avec une puissance de transmission variable, dans le but de déterminer une fonction de broadcast qui minimise le coût total tout en garantissant que chaque sommet du réseau est atteint.

Cependant, les applications réelles des réseaux complexes font face à des défis constants liés aux vulnérabilités sécuritaires et à la sensibilité des systèmes aux attaques ou aux pannes soudaines. C'est dans cette optique que s'inscrit la recherche scientifique que nous présentons dans ce mémoire, visant à améliorer l'efficacité de ce modèle et à le développer. Ainsi, nous avons élargi l'approche classique pour la transformer en une structure protégée et résiliente face aux intrusions, en introduisant le concept de *secure broadcast domination*.

Ce développement ne se limite pas à satisfaire aux conditions de couverture et de

portée, mais impose également des mécanismes de protection stricts qui garantissent la résilience du réseau. Ainsi, si un sommet dominant subit une attaque ou une défaillance fonctionnelle, la reconfiguration et la redistribution des rôles de broadcast se font automatiquement entre les autres sommets, préservant ainsi la stabilité et la sécurité globale du réseau sans interruption.

L'objectif de ce mémoire est l'étude du paramètre de *secure broadcast domination* dans les graphes, et d'analyser l'efficacité ainsi que la robustesse de ce modèle face aux pannes ou aux attaques. La notion de secure broadcast domination possède plusieurs domaines d'applications majeurs, citons notamment :

- **Sécurité et résilience des réseaux** : elle permet de modéliser la protection des infrastructures critiques, assurant que si un sommet émetteur subit une panne, les autres sommets se reconfigurent automatiquement pour maintenir le broadcast.
- **Optimisation du routage** : ce concept aide à concevoir des protocoles de communication sûrs où le coût total de transmission est minimal tout en garantissant une redondance sécurisée.

Ce mémoire est composé de quatre chapitres. Le premier chapitre est consacré aux notions fondamentales et aux terminologies de la théorie des graphes qui seront utiles pour la compréhension de ce travail. Dans le deuxième chapitre, nous présentons l'essentiel de la domination standard et de la broadcast domination classique. Le troisième chapitre est dédié à l'étude théorique du paramètre de la secure broadcast domination, incluant ses définitions et ses bornes. Enfin, dans le quatrième chapitre, nous proposons une approche heuristique pour estimer ce paramètre et nous évaluons les performances de nos algorithmes à travers des simulations numériques sur des réseaux complexes.

Concepts fondamentaux

Dans ce chapitre, nous présentons les définitions fondamentales ainsi que les notations nécessaires en théorie des graphes, indispensables à la compréhension de ce travail. Nous commençons par introduire les notions de base telles que les sommets, les arêtes, les chaînes et les cycles, tout en précisant la terminologie adoptée. Nous présentons ensuite les principaux paramètres des graphes, permettant de décrire leur structure et d'analyser leurs propriétés. Enfin, nous abordons certaines classes particulières de graphes qui serviront de cadre d'étude et d'illustration pour les résultats développés dans les chapitres suivants.

1.1 Notions et terminologie

Définition 1.1.1 Un graphe est un ensemble de relations entre un ou plusieurs objets. On l'utilise pour modéliser des réseaux tels que les circuits électriques et les réseaux de transport, ect.... . Chaque objet peut être représenté sous forme d'un point (sommet ou noeud), et on utilise des arêtes ou des arcs pour représenter les relations entre les sommets. Selon la nature de la relation, on distingue deux types de graphes : les graphes non orientés et les graphes orientés.

Ordre et taille d'un graphe

- On appelle **ordre d'un graphe** le nombre de ses sommets, c'est $|V|$.
- On appelle **taille d'un graphe** le nombre de ses arêtes, c'est $|E|$.

Graphe non orienté

Un **graphe non orienté** G est un couple $G = (V, E)$ tel que :

- V est un ensemble fini de *sommets* ;
- E est un ensemble de *couples non ordonnés* de sommets $(v_i v_j)$.

On mentionne parfois le nom du graphe pour éviter toute confusion, en écrivant $V(G)$ et $E(G)$. Une paire $(v_i v_j)$ est appelée une *arête*. On dit que les sommets v_i et v_j sont *adjacents*. L'arête $(v_i v_j)$ est dite *incidente* aux sommets v_i et v_j .

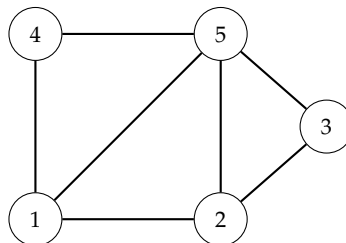


FIGURE 1.1 – Graphe non orienté

Graphe orienté

Un **graphe orienté** G est un couple $G = (V, E)$ tel que :

- V est un ensemble fini de sommets ;
- E est un ensemble de couples ordonnés de sommets $(v_i, v_j) \in V^2$.

Un couple (v_i, v_j) est appelé un arc, et est représenté graphiquement par $v_i \rightarrow v_j$. Le sommet v_i est le sommet initial ou origine, et v_j est le sommet terminal ou extrémité finale. L'arc $a = (v_i, v_j)$ est dit sortant de v_i et incident en v_j . Ainsi, v_j est un successeur de v_i , tandis que v_i est un prédécesseur de v_j . L'ensemble des successeurs d'un sommet $v_i \in V$ est noté : $\text{Succ}(v_i) = \{v_j \in V \mid (v_i, v_j) \in E\}$.

L'ensemble des prédécesseurs d'un sommet $v_i \in V$ est noté :

$\text{Pred}(v_i) = \{v_j \in V \mid (v_j, v_i) \in E\}$.

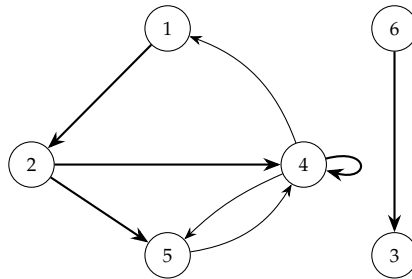


FIGURE 1.2 – Graphe orienté

Graphe simple et multigraphe

Un graphe est dit *simple* s'il ne comporte pas de boucle, et s'il ne comporte jamais plus d'une arête entre deux sommets. Un *multi-graphe* est un graphe qui peut avoir plusieurs arêtes entre deux mêmes sommets, ou des boucles.

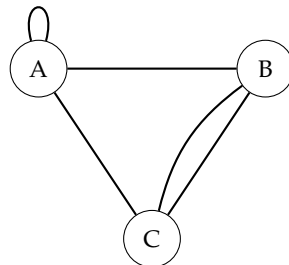


FIGURE 1.3 – Multigraphe

Graphe partiel et sous-graphe

Soit $G = (V(G), E(G))$ un graphe.

- On dit que $H = (V(H), E(H))$ est un sous-graphe de G si : $V(H) \subseteq V(G)$ et $E(H) \subseteq E(G)$, où chaque arête de $E(H)$ possède ses extrémités dans $V(H)$.
- On dit que H est un graphe partiel de G si : $V(H) = V(G)$ et $E(H) \subseteq E(G)$.

Soit $S \subseteq V(G)$ un sous-ensemble non vide de sommets du graphe G .

Le sous-graphe induit par S dans G , noté $G[S]$, est défini par : $V(G[S]) = S$ et $E(G[S]) = \{uv \in E(G) : u, v \in S\}$.

Autrement dit, $G[S]$ contient tous les sommets de S ainsi que toutes les arêtes de G dont les deux extrémités appartiennent à S .

Graphes connexes et connexité

Un graphe G est *connexe* si pour toute paire de sommets $v_1, v_n \in V$, il existe une chaîne entre v_1 et v_n : on dit alors que les sommets v_1 et v_n sont connectés. Un graphe non connexe se décompose en **composantes connexes** du graphe.

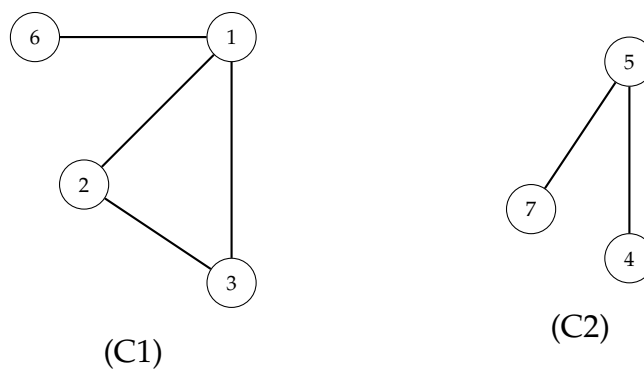


FIGURE 1.4 – Graphe non connexe mais les composantes (C1) (C2) sont connexes

Degré d'un graphe

Pour un sommet v d'un graphe G , le **degré** de v est le nombre d'arêtes incidentes à ce sommet, noté $d_G(v)$, (C'est le nombre de sommets adjacents au sommet v).

- Un sommet de degré 0 ($d_G(v) = 0$) est dit **sommet isolé**.
- Un sommet de degré 1 ($d_G(v) = 1$) est dit **sommet pendent**, et son voisin est dit **support**.
- Le degré minimum du graphe G est le plus petit degré d'un sommet, noté $\delta(G)$.
- Le degré maximum est le plus grand degré d'un sommet de G , noté $\Delta(G)$.

Par exemple, pour le graphe G de la figure 1.1, le degré minimum de G est $\delta(G) = 2$ et le degré maximum $\Delta(G) = 4$.

Dans un graphe orienté, on a :

- Le **demi-degré extérieur** d'un sommet v est égal au nombre d'arcs ayant le sommet v comme extrémité initiale. On le note : $d^+(v) = |\{u \in E \mid I(u) = v\}|$
où $I(u)$ désigne l'extrémité initiale de l'arc u .
- Le **demi-degré intérieur** d'un sommet v est égal au nombre d'arcs ayant le sommet v comme extrémité terminale. On le note : $d^-(v) = |\{u \in E \mid T(u) = v\}|$
où $T(u)$ désigne l'extrémité terminale de l'arc u .
- Le **degré total** d'un sommet v est le nombre total d'arcs ayant v comme extrémité initiale ou terminale. On le note : $d(v) = d^+(v) + d^-(v)$

Par exemple, pour le graphe G de la figure 1.2 : $d^+(2) = 2$, $d^-(2) = 1$, $d(2) = d^+(2) + d^-(2) = 3$.

Voisinage

Étant donné un sommet v d'un graphe $G = (V, E)$, on appelle **voisinage** de v et on note $N(v)$ l'ensemble des sommets *adjacents* à v .

Formellement :

- $N(v) = \{u \in V \mid (u, v) \in E\}$. Les éléments de $N(v)$ sont appelés les *voisins ouverts* du sommet v .
- $N[v] = N(v) \cup \{v\}$. Les éléments de $N[v]$ sont appelés les *voisins fermés* du sommet v .

Par exemple, pour le graphe G de la figure 1.1, le voisinage ouvert du sommet 1 est $N(1) = \{5, 4, 2\}$ et son voisinage fermé est $N[1] = \{1, 5, 4, 2\}$.

Dans un **graphe orienté**, on distingue deux types de voisinages :

- Le **voisinage extérieur** de v , noté $N^+(v)$, est l'ensemble des sommets vers lesquels partent les arcs issus de v .
- Le **voisinage intérieur** de v , noté $N^-(v)$, est l'ensemble des sommets dont les arcs aboutissent à v .

Par exemple, pour le graphe G de la figure 1.2, le voisinage extérieur $N^+(2) = \{4, 5\}$ et le voisinage intérieur $N^-(2) = \{1\}$.

Chaîne (Chemin)

Une **chaîne** dans un graphe $G = (V, E)$ est une suite ayant pour éléments alternativement des sommets et des arêtes, commençant et se terminant par un sommet, et telle que chaque arête est encadrée par ses extrémités. On note : $(x_0 e_1 x_1 e_2 \dots e_k x_k)$.

où k est un entier ≥ 0 , $x_i \in V$ pour $i = 0, \dots, k$, et $e_j \in E$ pour $j = 1, \dots, k$, et pour tout $i = 0, \dots, k-1$, les sommets x_i et x_{i+1} sont les extrémités de e_{i+1} .

L'entier k est appelé la **longueur de la chaîne**. (Dans le cas des graphes orientés, une telle suite est appelée un **chemin**.)

Chaîne simple

La chaîne simple est une chaîne qui n'utilise pas deux fois la même arête.

Chaîne eulérienne

La chaîne eulérienne dans un graphe $G = (V, E)$ est une chaîne simple qui passe par toutes les arêtes du graphe une et une seule fois.

Chaîne hamiltonienne

La chaîne hamiltonienne dans un graphe $G = (V, E)$ est une chaîne simple qui passe par tous les sommets du graphe une et une seule fois.

Représentation d'un graphe

En théorie des graphes, il existe plusieurs façons fondamentales de représenter un graphe $G = (V, E)$, où V est l'ensemble des sommets et E est l'ensemble des arêtes ou arcs. Ces représentations permettent de modéliser un graphe soit mathématiquement, soit sous forme informatique. Parmi les représentations les plus utilisées, on trouve :

- **Représentation par matrice d'adjacence :**

La **matrice d'adjacence** est définie pour un graphe d'ordre $n = |V|$ en numérotant les sommets de 1 à n . On forme alors une matrice $A = (a_{ij}) \in \{0, 1\}^{n \times n}$ telle que :

$$a_{ij} = \begin{cases} 1, & \text{s'il existe une arête ou un arc entre le sommet } i \text{ et le sommet } j, \\ 0, & \text{sinon.} \end{cases}$$

Si le graphe est non orienté, la matrice A est symétrique. Cette représentation est particulièrement utile pour savoir rapidement si deux sommets sont connectés ou non. Par exemple, pour le graphe G de la figure 1.1, La **matrice d'adjacence** associée à ce graphe est donnée par :

$$A = \begin{pmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \end{pmatrix}$$

- **Représentation par matrice d'incidence :**

le graphe G possède n sommets et m arêtes, la matrice d'incidence $M = (m_{ij})$ est une matrice de taille $n \times m$ définie par :

$$m_{ij} = \begin{cases} 1, & \text{si le sommet } i \text{ est incident à l'arête } e_j, \\ 0, & \text{sinon.} \end{cases}$$

Dans le cas d'un graphe orienté :

$$m_{ij} = \begin{cases} 1, & \text{si } e_j \text{ sort du sommet } i, \\ -1, & \text{si } e_j \text{ entre dans le sommet } i, \\ 0, & \text{sinon.} \end{cases}$$

1.2 Paramètres des graphes

Distance

La distance entre deux sommets dans un graphe non orienté est la **longueur du plus petite chaîne** reliant ces deux sommets.

Par exemple, pour le graphe G de la figure 1.1 : $d(1,2) = 1$, $d(1,3) = 2$, $d(1,4) = 1$, $d(1,5) = 1$, $d(2,3) = 1$, $d(2,4) = 2$, $d(2,5) = 1$, $d(3,4) = 2$, $d(3,5) = 1$, $d(4,5) = 1$.

Excentricité d'un sommet

L'**excentricité** d'un sommet u dans un graphe G est la plus grande distance entre le sommet u et n'importe quel autre sommet v de G . Autrement dit : $e(u) = \max_{v \in V} d(u, v)$.

Par exemple, le graphe G de la figure 1.1 : $e(1) = 2$, $e(2) = 2$, $e(3) = 2$, $e(4) = 2$, $e(5) = 1$.

Rayon d'un graphe

Le **rayon** d'un graphe G , noté $\text{Rad}(G)$, est la plus petite excentricité parmi tous les sommets du graphe : $\text{Rad}(G) = \min_{u \in V} e(u)$.

Par exemple, le rayon du graphe de la figure 1.1 est : $\text{Rad}(G) = \min\{2, 2, 2, 2, 1\} = 1$. Le sommet 5 est donc le sommet le plus central, car il possède l'excentricité minimale.

Diamètre d'un graphe

Le **diamètre** d'un graphe G , noté $\text{Diam}(G)$, est la plus grande excentricité parmi tous les sommets du graphe : $\text{Diam}(G) = \max_{u \in V} e(u)$.

Par exemple, le diamètre du graphe de la figure 1.1 : $\text{Diam}(G) = \max\{2, 2, 2, 2, 1\} = 2$.

Nombre de stabilité

Le **nombre de stabilité** d'un graphe G , noté $\alpha(G)$, est la taille maximum d'un ensemble de sommets deux à deux non adjacents. Autrement dit, c'est le plus grand cardinal d'un ensemble stable de G .

$$\alpha(G) = \max\{|S| \mid S \subseteq V(G), \forall u, v \in S, (u, v) \notin E(G)\}.$$

- Un **stable maximal** est un ensemble stable qui ne peut pas être agrandi sans perdre la stabilité. Autrement dit, pour tout $v \in V \setminus S$, l'ensemble $S \cup \{v\}$ n'est pas stable.
- Un **stable maximum** est un ensemble stable de taille maximale, c'est-à-dire de cardinal égal à $\alpha(G)$, le *nombre d'indépendance* du graphe.

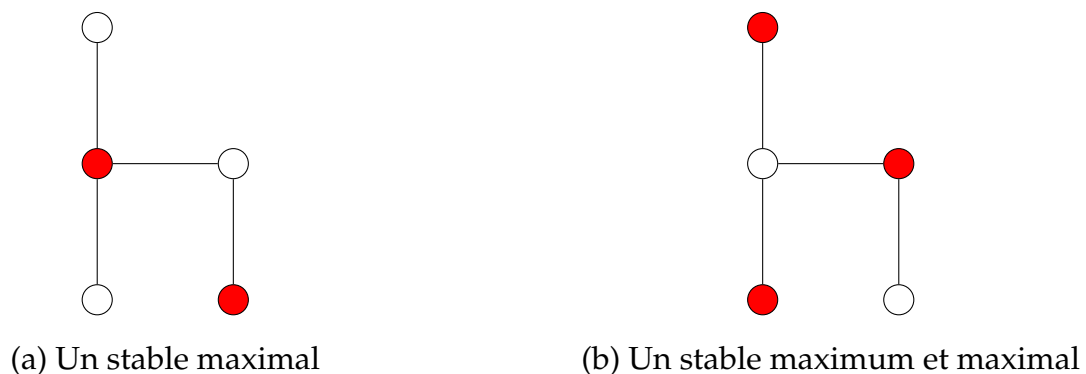


FIGURE 1.5 – Stable maximal et stable maximum

Nombre chromatique (Coloration)

Le **nombre chromatique** d'un graphe G , noté $\chi(G)$, est le plus petit nombre de couleurs nécessaires pour colorier les sommets de G de telle sorte que deux sommets adjacents aient toujours des couleurs différentes. Une telle attribution de couleurs est appelée une **coloration propre** du graphe.

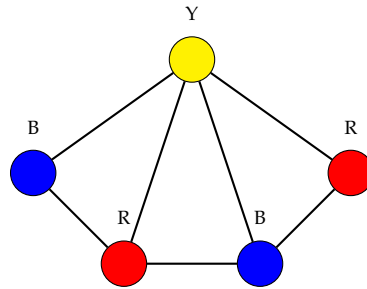


FIGURE 1.6 – Coloration propre des sommets d'un graphe G

Définition 1.2.1 Une **clique** d'un graphe $G = (V, E)$ est un sous-ensemble de sommets $C \subseteq V$ tel que chaque paire de sommets distincts de C est reliée par une arête. Autrement dit :

$$\forall u, v \in C, u \neq v \Rightarrow (uv) \in E.$$

Nombre de clique

Le **nombre de clique** d'un graphe G , noté $\omega(G)$, est la taille maximum d'une clique de G . Autrement dit : $\omega(G) = \max\{|C| \mid C \subseteq V(G), C \text{ est une clique de } G\}$.

1.3 Quelques classes de graphes

Graphe trivial

Un *graphe trivial* est un graphe qui ne contient qu'un seul sommet et aucune arête. Autrement dit : $G = (V, E)$ où $V = \{v_1\}$, $E = \emptyset$.

Graphe planaire

Un graphe est dit **planaire** s'il peut être plongé dans le **plan** sans que deux arêtes se croisent.

Graphe complet

Un **graphe complet** est un **graphe simple non orienté** tel que toute paire de sommets est **adjacente**. Le graphe complet à n sommets sera noté K_n .

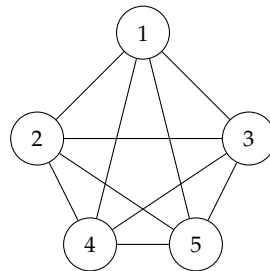


FIGURE 1.7 – Graphe complet K_5

Graphe réflexif

On appelle **graphe réflexif** un graphe $G = (V, E)$ tel que chaque sommet possède une boucle, c'est-à-dire : $\forall v \in V, (v, v) \in E$

Graphe eulérien

Un *graphe eulérien* est un graphe qui possède un cycle eulérien.

Graphe semi-eulérien

Un *graphe semi-eulérien* est un graphe qui possède une chaîne eulérienne.

Graphe hamiltonien

Un *graphe hamiltonien* est un graphe qui possède un cycle hamiltonien.

Graphe semi-hamiltonien

Un *graphe semi-hamiltonien* est un graphe qui possède une chaîne hamiltonienne.

Cycle

Un **cycle** dans un graphe G non orienté est une chaîne de longueur simple et fermée. C'est donc une suite de la forme : $(x_0 e_1 x_1 e_2 \dots e_k x_0)$ où k est un entier ≥ 1 , les x_i et les

e_j étant définis comme précédemment. L'entier k est la **longueur du cycle**.

Un cycle est dit **élémentaire** si les sommets x_i pour $i = 0, \dots, k - 1$ sont distincts deux à deux.

Tout cycle peut se décomposer en cycles élémentaires deux à deux disjoints relativement aux arêtes.

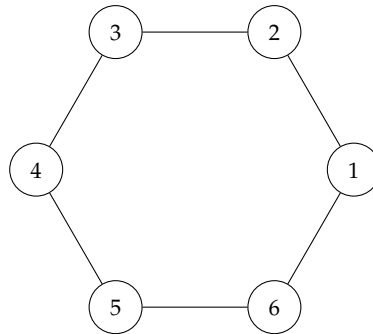


FIGURE 1.8 – Cycle C_6

Cycle eulérien

Un **cycle eulérien** dans un graphe non orienté $G = (V, E)$ est un cycle simple qui passe par toutes les arêtes du graphe une et une seule fois.

Cycle hamiltonien

Un **cycle hamiltonien** dans un graphe non orienté $G = (V, E)$ est un cycle simple qui passe par tous les sommets du graphe une et une seule fois.

Circuit

Un **circuit** dans un graphe orienté $G = (V, E)$ est un chemin fermé c'est-à-dire une suite de sommets $v_1, v_2, \dots, v_n$ telle que : $[v_1 = v_n]$, et pour tout $i = 1, 2, \dots, n - 1$, les sommets v_i et v_{i+1} sont adjacents. De plus, aucun sommet (sauf le premier et le dernier) n'est répété.

Circuit eulérien

Un **circuit eulérien** dans un graphe orienté est un circuit qui passe une et une seule fois par tout les arcs.

Circuit hamiltonien

Un **circuit hamiltonien** dans un graphe orienté est un circuit que passe une et une seule fois par tout les sommets.

Arbre

L'arbre est une sous-classe très étudiée des graphes bipartis et des graphes cordaux. Un arbre est un graphe connexe sans cycle, noté T , et comportant exactement $n - 1$ arêtes, et n le nombre de sommets.

Dans un arbre, on distingue trois types de sommets :

- **Les feuilles** : ce sont les sommets de degré 1.
- **Les noeuds internes** : ce sont les sommets de degré strictement supérieur à 1.
- **La racine** : dans un arbre enraciné, un sommet particulier est désigné comme racine.

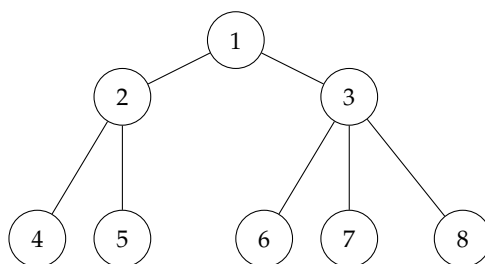
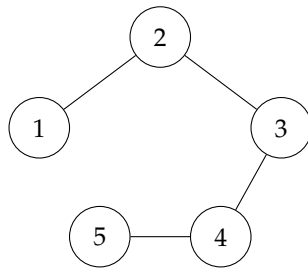


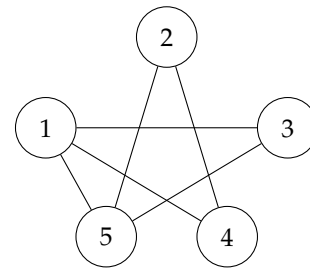
FIGURE 1.9 – L'arbre contient une racine 1, des sommets internes 2 et 3, et 4, 5, 6, 7, 8 des feuilles.

Graphe complémentaire

Soit $G = (V, E)$ un graphe simple non orienté. On appelle **graphe complémentaire** de G , noté $\bar{G} = (V, \bar{E})$, le graphe défini sur le même ensemble de sommets V , tel que deux sommets distincts sont adjacents dans $\bar{G}$ si et seulement s'ils ne le sont pas dans G . Autrement dit, le graphe complémentaire inverse la relation d'adjacence : toute arête existant dans G est absente dans $\bar{G}$, et inversement



G : graphe original



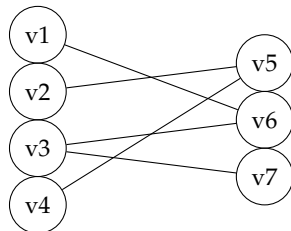
$\bar{G}$: graphe complément de G

FIGURE 1.10 – G et son complément $\bar{G}$

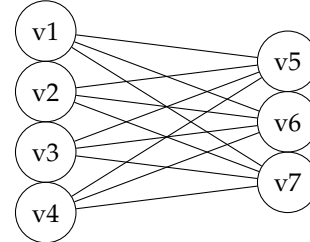
Graphe biparti et graphe biparti complet

Un graphe G est **biparti** si l'ensemble de ses sommets peut être partitionné en deux parties V_1 et V_2 telles que le sous-graphe induit par V_1 ne contient aucune arête, et celui induit par V_2 non plus. Autrement dit, le graphe biparti ne contient pas de cycle de longueur impaire.

Le **graphe biparti complet** est un graphe biparti dans lequel chaque sommet de V_1 est relié à *tout* sommet de V_2 , et chaque sommet de V_2 est relié à *tout* sommet de V_1 . Il est noté $K_{m,n}$ avec m l'ordre de V_1 (c'est-à-dire $|V_1| = m$) et n l'ordre de V_2 (c'est-à-dire $|V_2| = n$).



(a) Graphe biparti



Graphe biparti complet $K_{4,3}$

FIGURE 1.11 – Graphe biparti et son graphe biparti complet $K_{4,3}$

Graphe symétrique et antisymétrique

On dit qu'un graphe orienté est **symétrique** si, pour tout couple de sommets v_i, v_j ,

$$(v_i, v_j) \in E \implies (v_j, v_i) \in E.$$

Autrement dit : dès qu'il existe un arc allant de v_i vers v_j , il existe aussi l'arc inverse de v_j vers v_i . On dit qu'un graphe orienté est **antisymétrique** si, pour tout couple de sommets distincts v_i, v_j ,

$$(v_i, v_j) \in E \implies (v_j, v_i) \notin E.$$

Autrement dit : il n'existe pas deux arcs de sens contraire entre deux sommets différents (les seules éventuelles arcs réciproques sont des boucles (v_i, v_i)). Dans le cas d'un **graphe non orienté**, la notion de symétrie/antisymétrie n'a pas de sens logique, puisque chaque arête est naturellement non orientée (équivalente aux deux orientations (v_i, v_j) et (v_j, v_i)). Un graphe orienté peut donc être symétrique, antisymétrique, ou n'être ni l'un ni l'autre.

Domination et broadcasts domination dans les graphes

Dans ce chapitre, nous présentons les notions fondamentales de la domination dans les graphes, en introduisant d'abord la définition de la domination ainsi que quelques paramètres invariants associés. Nous abordons ensuite les différentes variantes de la domination, en mettant en évidence leurs propriétés et leurs spécificités.

Par la suite, nous nous intéressons à la notion de *broadcast domination*, en exposant ses principes de base ainsi que ses invariants propres. Enfin, nous présentons certaines relations, sous forme d'égalités et d'inégalités, reliant le nombre de domination classique $\gamma(G)$ et le nombre de *broadcast domination* $\gamma_b(G)$.

2.1 Domination dans les graphes

Ensembles dominants

Soit $G = (V, E)$ un graphe simple. Un sous-ensemble de sommets D de V est un dominant si tout sommet de $V \setminus D$ est adjacent à au moins un sommet de D . Le cardinal minimum d'un ensemble dominant de G appelé nombre de domination est noté par $\gamma(G)$ et le cardinal maximum d'un ensemble dominant minimal de G appelé nombre de domination supérieur est noté par $\Gamma(G)$.

Dans la littérature, il existe d'autres définitions équivalentes aux ensembles dominants dans les graphes. En voici des exemples :

- Un ensemble $D \subseteq V$ est un dominant si pour tout $v \in V$, $|N[v] \cap D| \geq 1$,
- Un ensemble $D \subseteq V$ est un dominant si pour tout $v \in V \setminus D$, $N(v) \cap D \neq \emptyset$,
- Un ensemble $D \subseteq V$ est un dominant si $N[D] = V$.

Par exemple pour le graphe de la figure 2.1 : $\mathcal{D} = \{\{a, f\}, \{b, e\}, \{c, d\}, \{a, d, e\}, \{f, b, c\}\}$, sont ensemble dominants. C'est ce que ces ensemble dominants sont tous minimal.

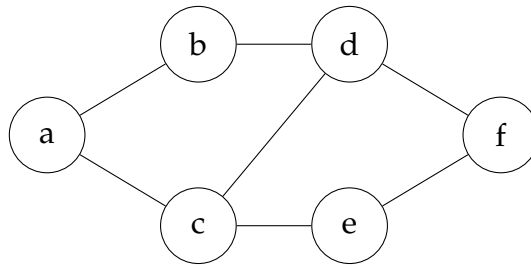


FIGURE 2.1 – Un graphe G avec $\gamma(G) = 2$ et $\Gamma(G) = 3$

Adjacents privés

Soit $G = (V, E)$ un graphe et $D \subseteq V$ un ensemble dominant de G .

Pour tout sommet $v \in D$, on appelle **adjacents privés** de v (par rapport à D) l'ensemble des sommets de V qui sont adjacents à v et à aucun autre sommet de D , c'est-à-dire : $Priv(v) = \{u \in V \mid u \in N(v) \text{ et } u \notin N(D \setminus \{v\})\}$.

Bornes sur le nombre de domination $\gamma(G)$

Étant donné que la détermination du nombre de domination $\gamma(G)$ d'un graphe quelconque est un problème NP-complet [30, 4], il est souvent difficile d'en calculer la

valeur exacte. Ainsi, il est important de rechercher des encadrements de ce paramètre, notamment des bornes supérieures et inférieures.

Dans cette section, nous présentons quelques résultats concernant le nombre de domination $\gamma(G)$, en l'étudiant en fonction de différents paramètres du graphe G , tels que son ordre, sa taille, les degrés de ses sommets (minimum et maximum), ainsi que son diamètre.

Il est clair que le nombre de sommets n du graphe G constitue une borne supérieure pour $\gamma(G)$. Cette borne est atteinte lorsque tous les sommets du graphe sont isolés, tandis que la borne inférieure est obtenue lorsqu'il existe un sommet de degré $n - 1$, c'est-à-dire adjacent à tous les autres sommets du graphe.

Théorème 2.1.1 [30] *Si G est un graphe sans sommets isolés, alors : $\gamma(G) \leq \frac{n}{2}$.*

Théorème 2.1.2 [23] *Pour un graphe G d'ordre n , connexe et dont le degré minimum des sommets est supérieur ou égal à 3, on a : $\gamma(G) \leq \frac{3n}{8}$.*

La chaîne P_8 vérifie $\gamma(P_8) = 3$, ce qui montre que la borne donnée dans le théorème peut être atteinte.

Walikar, Acharya et Sampathkumar ont également proposé une autre borne pour $\gamma(G)$ dépendant de l'ordre n et du degré maximum $\Delta(G)$.

Théorème 2.1.3 [4] *Soit G un graphe non orienté simple, tel que $|V(G)| = n$ et $|E(G)| = m$, alors : $n - m \leq \gamma(G) \leq n - \Delta(G)$.*

Théorème 2.1.4 [24] *Pour tout graphe G d'ordre n , on a :*

$$1. \gamma(G) + \gamma(\overline{G}) \leq n + 1$$

Théorème 2.1.5 [23] *Pour tout graphe G d'ordre n et de degré maximum $\Delta(G)$, on a :*

$$\frac{n}{\Delta(G)+1} \leq \gamma(G).$$

Théorème 2.1.6 [1] *Soit G un graphe sans sommets isolés tel que $\gamma(G) \geq 3$, alors : $\gamma(G) \leq \frac{n+1-\delta(G)}{2}$.*

Une borne plus précise, dépendant à la fois de l'ordre de G et de son degré minimum pour les graphes sans sommets isolés, a été démontrée par Alon et Spencer.

Théorème 2.1.7 [25] *Si G est un graphe tel que $\text{diam}(G) = 2$, alors : $\gamma(G) \leq \delta(G)$.*

Voici quelques exemples de graphes G pour lesquels on observe que $\gamma(G) \leq \delta(G)$, notamment les graphes bipartis complets $K_{m,n}$ avec $m, n \geq 2$, la chaîne P_3 , ainsi que les cycles C_n pour $n = 4, 5, 6$. Une relation intéressante entre le diamètre d'un graphe G et le nombre de domination de son complémentaire $\overline{G}$ a été étudiée par Brigham, Chinn et Dutton.[9]

Valeurs du nombre de domination pour certaines classes de graphes

Proposition 2.1.1 [23] Pour tout entier $n \geq 3$, on a : $\gamma(P_n) = \left\lceil \frac{n}{3} \right\rceil$.

Proposition 2.1.2 [23] Pour tout entier $n \geq 3$, on a : $\gamma(C_n) = \left\lceil \frac{n}{3} \right\rceil$.

Proposition 2.1.3 [23] Pour tout graphe complet K_n , on a : $\gamma(K_n) = 1$.

Proposition 2.1.4 [23] Pour tout graphe biparti complet $K_{m,n}$ avec $m, n \geq 2$, on a : $\gamma(K_{m,n}) = 2$.

2.2 Les variantes de la domination

Domination indépendante

Soit $G = (V, E)$ un graphe. Un ensemble $S \subseteq V(G)$ est appelé *ensemble dominant indépendant* (ou *domination indépendante*) si : $\forall v \in V \setminus S, \exists u \in S$ tel que $uv \in E$, et $\forall u, v \in S, uv \notin E(G)$. Le nombre de domination indépendante de G , noté $i(G)$, est défini par : $i(G) = \min\{|S| : S \text{ est une domination indépendante de } G\}$.

par exemple la chaine suivant : $S = \{2, 5\}$ où $i(P_5) = 2$.

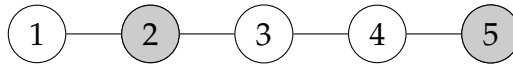


FIGURE 2.2 – Ensemble dominant indépendant

Proposition 2.2.1 [5] Pour un graphe G d'ordre n et de degré maximum Δ , on a :

$$\frac{n}{1+\Delta} \leq i(G) \leq n - \Delta.$$

Théorème 2.2.1 [8] Soit G un graphe sans sommets isolés d'ordre n , alors :

$$i(G) \leq n + 2 - \gamma(G) - \left\lceil \frac{n}{\gamma(G)} \right\rceil.$$

Théorème 2.2.2 [20] Soit G un graphe sans sommets isolés d'ordre n , alors : $i(G) \leq n + 2 - 2\sqrt{n}$.

K-domination

Soit D un ensemble de sommets d'un graphe $G = (V, E)$. On dit que D est *k-dominant* si chaque sommet de $V \setminus D$ est adjacent à au moins k sommets de D .

Le *nombre de k-domination* d'un graphe G , noté $\gamma_k(G)$, est la plus petite cardinalité d'un ensemble *k-dominant* de G .

Proposition 2.2.2 [21] Pour tout graphe G , on a : $\gamma_k(G) \geq \frac{kn}{\Delta+k}$. Où Δ désigne le degré maximum de G .

Proposition 2.2.3 [14] Si $\delta(G) \geq k$, alors : $\gamma_k(G) \leq \frac{kn}{k+1}$.

Nous abordons dans cette partie un cas particulier de la k -domination, appelé la domination double.

Domination double

Soit $G = (V, E)$ un graphe simple. Un sous-ensemble $S \subseteq V$ est un *ensemble dominant double* de G si, pour tout sommet $v \in V$, on a : $|[N[v] \cap S]| \geq 2$, c'est-à-dire :

- si $v \in S$, alors v a au moins un voisin dans S ;
- si $v \in V \setminus S$, alors v a au moins deux voisins dans S .

Le *nombre de domination double*, noté $\gamma_{\times 2}(G)$, est le cardinal minimum d'un ensemble dominant double de G .

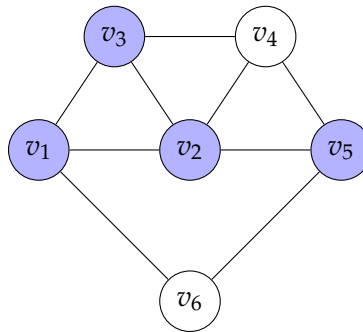


FIGURE 2.3 – Ensemble dominant double $S = \{v_1, v_2, v_3, v_5\}$, est ce que S est minimum $\gamma_{\times 2}(G) = 4$

Théorème 2.2.3 [22] Soit G un graphe sans sommets isolés ayant n sommets et m arêtes. Alors :

- $\gamma_{\times 2}(G) \geq \gamma(G) + 1$;
- Si G admet deux $\gamma(G)$ -ensembles disjoints, alors $\gamma_{\times 2}(G) \leq 2\gamma(G)$;
- $2 \leq \gamma_{\times 2}(G) \leq n$;
- $\gamma_{\times 2}(G) \geq \frac{4n - 2m}{3}$;
- $\gamma_{\times 2}(K_n) = 2$ pour tout $n \geq 2$;
- $\gamma_{\times 2}(C_n) = \left\lceil \frac{2n}{3} \right\rceil$.

Théorème 2.2.4 [12] Pour tout graphe G sans sommets isolés, on a : $\gamma_{\times 2}(G) \leq n + 1 - \delta$.

Corollaire 2.2.1 [12] Soit G un graphe d'ordre n tel que $G \neq C_5$ et de degré minimum $\delta(G) \geq 2$, alors on a : $\gamma_{\times 2}(G) \leq \left\lfloor \frac{11n}{14} \right\rfloor$.

Théorème 2.2.5 [6] Soit T un arbre, alors : $\gamma_{\times 2}(T) = 2\gamma(T)$. Si et seulement si T possède deux $\gamma(G)$ -ensembles disjoints.

Théorème 2.2.6 [6] Soit T une chenille, alors : $\gamma_{\times 2}(T) = 2\gamma(T)$.

Si et seulement si T est la chaîne P_2 ou bien T est tel que chaque sommet support est adjacent à exactement un seul sommet pendant et la distance entre deux sommets supports consécutifs est congrue à 1 ou à 2 modulo 3.

Théorème 2.2.7 [22] Si G est un graphe tel que $\text{diam}(G) = 1$, alors on a : $\gamma_{\times 2}(G) = 2$.

Théorème 2.2.8 [22] Si G est un graphe ayant $\text{diam}(G) = 2$, alors on a : $\gamma_{\times 2}(G) \leq \Delta + \delta$.

On appelle *maille* d'un graphe G , la longueur du plus petit cycle dans G . La maille d'un graphe G est notée par $g(G)$.

Théorème 2.2.9 [22] Pour tout graphe connexe G d'ordre n tel que $\delta \geq 2$, on a :

$$\gamma_{\times 2}(G) \leq n + \frac{1 - \text{diam}(G)}{3}.$$

Domination totale

Un ensemble S de sommets d'un graphe fini et non orienté $G = (V, E)$ est un **ensemble domination totale** si chaque sommet de V est adjacent à au moins un sommet de S .

Le **nombre de domination totale** de G , noté $\gamma_t(G)$, est le plus petit nombre de sommets dans un ensemble domination totale.

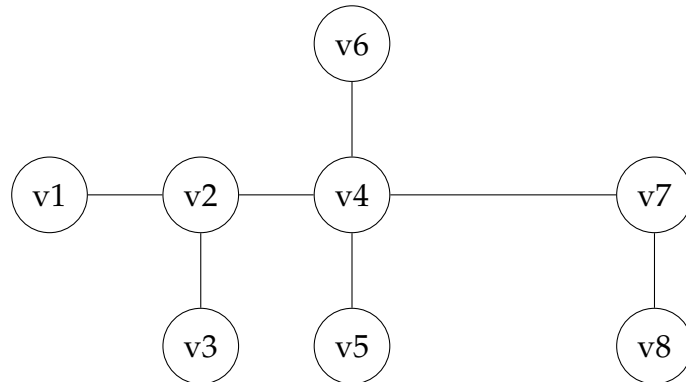


FIGURE 2.4 – Ensemble dominant total : $S = \{v_2, v_4, v_7\}$

Théorème 2.2.10 [13] Pour tout graphe connexe G et d'ordre $n \geq 3$, on a : $\gamma_t(G) \leq \frac{2n}{3}$.

Théorème 2.2.11 [8] Soit G un graphe connexe d'ordre n , $n \geq 3$, alors : $\gamma_t(G) \leq \frac{2n}{3}$. Si et seulement si G est un cycle C_3 ou C_6 , ou la 2-couronne d'un graphe connexe.

Théorème 2.2.12 [11] Si T est un arbre d'ordre $n \geq 3$ avec s sommets supports, alors :

- $\gamma_t(T) \leq \frac{n+s}{2}$;
- $\gamma_{pr}(T) \leq \frac{n+2s-1}{2}$.

Théorème 2.2.13 [26] Pour $G \in \{P_n, C_n\}$ avec $n \geq 3$, on a : $\gamma_t(G) = \left\lfloor \frac{n}{2} \right\rfloor + \left\lceil \frac{n}{4} \right\rceil - \left\lfloor \frac{n}{4} \right\rfloor$.

Théorème 2.2.14 [12] Si $G \notin \{C_3, C_5, C_6, C_{10}\}$ est un graphe d'ordre $n \geq 2$, alors : $\gamma_t(G) \leq \frac{4n}{7}$.

Fonction de domination romaine

Soit $G = (V, E)$ un graphe. Une **fonction de domination romaine (RDF)** sur G est une fonction $f : V \rightarrow \{0, 1, 2\}$ qui satisfait la condition suivante : Tout sommet $u \in V$ tel que $f(u) = 0$ est adjacent à au moins un sommet $v \in V$ avec $f(v) = 2$.

Le **poids** de la RDF f est défini par : $f(V) = \sum_{v \in V} f(v)$.

Le **nombre de domination romaine** d'un graphe G , noté $\gamma_R(G)$, est le poids minimum parmi toutes les RDF sur G : $\gamma_R(G) = \min\{f(V) \mid f \text{ est une RDF sur } G\}$.

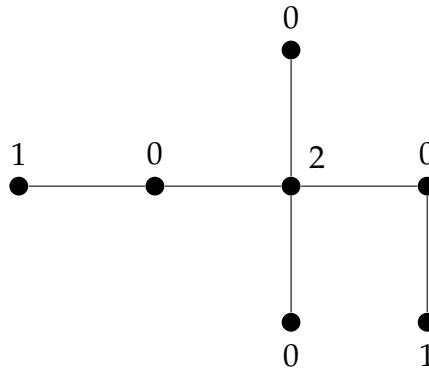


FIGURE 2.5 – Un graphe G avec $\gamma_R(G) = 4$

Lemme 2.2.1 [27] Pour tout graphe G , on a : $\gamma(G) \leq \gamma_r(G) \leq \gamma_R(G) \leq 2\gamma(G)$.

Lemme 2.2.2 [27] Pour tout entier $n \geq 1$, on a : $\gamma_r(P_n) = \gamma_r(C_n) = \left\lceil \frac{3n}{7} \right\rceil$.

Lemme 2.2.3 [33] Pour tout arbre T tel que $\gamma_r(T) = \gamma(T)$, l'arbre T ne contient aucun sommet support fort (strong support vertex). Autrement dit, pour tout $v \in V(T)$, on a $v \notin SS(T)$.

Théorème 2.2.15 [10] Soit T un arbre à n sommets, avec $n \geq 3$, alors : $\gamma_R(G) \leq \frac{4n}{5}$.

Domination sécurisée

Soit $G = (V(G), E(G))$ un graphe, un ensemble $S \subseteq V(G)$ est appelé *ensemble dominant sécurisé* si : $\forall u \in V(G) \setminus S, \exists v \in S$ tel que $u \in N(v)$, et $(S \setminus \{v\}) \cup \{u\}$ est un ensemble dominant de G .

Le nombre de domination sécurisée est noté $\gamma_s(G)$.

Domination sécurisée sur certaines classes de graphes

Théorème 2.2.16 [15] Pour les chaînes P_n et les cycles C_n , on a : $\gamma_s(P_n) = \gamma_s(C_n) = \left\lceil \frac{3n}{7} \right\rceil$

Par exemple dans la chaîne P_{10} , $\gamma_s(P_{10}) = 5$

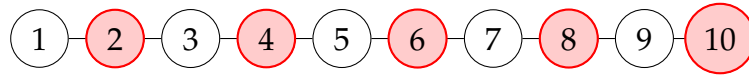


FIGURE 2.6 – La chaîne P_{10} avec $S = \{2, 4, 6, 8, 10\}$.

Proposition 2.2.4 [15] Pour le graphe complet K_n avec $n \geq 1$, $\gamma_s(K_n) = 1$.

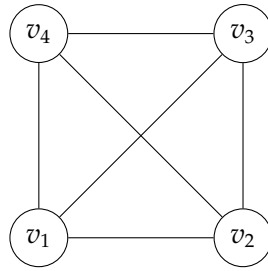


FIGURE 2.7 – Le graphe complet K_4 avec $S = \{v_2\}$

Proposition 2.2.5 [15] Pour le graphe biparti complet $K_{p,q}$ avec $1 \leq p \leq q$:

$$\gamma_s(K_{p,q}) = \begin{cases} q & \text{si } p = 1 \\ 2 & \text{si } p = 2 \\ 3 & \text{si } p = 3 \\ 4 & \text{si } p \geq 4 \end{cases}$$

Bornes sur le nombre de domination sécurisée $\gamma_s(G)$

Proposition 2.2.6 [23]. Pour tout graphe G , on a : $\gamma(G) \leq \gamma_s(G)$.

Théorème 2.2.17 [32] Soit G un graphe d'ordre n et de degré maximum $\Delta(G)$, alors :

$$\gamma_s(G) \geq \frac{n}{\Delta(G)+1}.$$

Proposition 2.2.7 [14] Pour tout graphe G sans sommet isolé, on a : $\gamma(G) \leq \gamma_s(G) \leq \gamma_t(G)$.

Proposition 2.2.8 [23] Soit G un graphe d'ordre n , alors : $\gamma_s(G) \leq n$.

Théorème 2.2.18 [15] Soit G un graphe sans triangle d'ordre n et de degré maximum Δ , alors : $\gamma_s(G) \geq \frac{n(2\Delta-1)}{\Delta^2+2\Delta-1}$, la borne est atteinte pour une infinité de graphes.

Théorème 2.2.19 [15] Soit G un graphe sans K_4 d'ordre n et de degré maximum $\Delta \geq 2$, alors : $\gamma_s(G) \geq \frac{n(2\Delta-3)}{\Delta^2+2\Delta-5}$, la borne est atteinte pour une infinité de graphes.

2.3 Broadcast domination

Soit $G = (V, E)$ un graphe simple, une fonction $f : V \rightarrow \{0, 1, \dots, \text{diam}(G)\}$ telle que : $\forall v \in V, f(v) \leq e(v)$, où $e(v)$ désigne l'excentricité du sommet v , est appelée un *broadcast* sur G .

Le f -voisinage fermé d'un sommet v est défini par : $N_f[v] = \{u \in V \mid d(u, v) \leq f(v)\}$.

L'ensemble des sommets ayant une valeur de broadcast positive est donné par : $V_f^+ = \{v \in V \mid f(v) > 0\}$.

Un broadcast f est dit *dominant* si : $\forall u \in V, \exists v \in V_f^+$ tel que $d(u, v) \leq f(v)$.

Le coût du broadcast f est défini par : $\sigma(f) = \sum_{v \in V} f(v)$.

Le nombre de broadcast-domination de G est défini par :

$\gamma_b(G) = \min\{\sigma(f) \mid f \text{ est un broadcast dominant sur } G\}$.

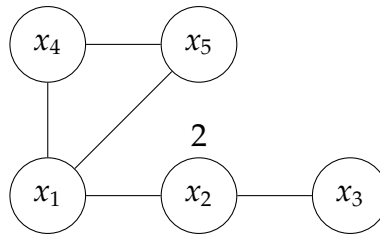


FIGURE 2.8 – L'ensemble broadcast est $V_f^+ = \{x_2\}$ et $\gamma_b(G) = 2$.

Théorème 2.3.1 [17] Soit G un graphe connexe, alors $\gamma_b(G) = \gamma(G)$ si et seulement s'il existe un broadcast dominant f de coût minimum sur G telle que : V_f^+ est un ensemble dominant. En effet, soit $G(V, E)$ un graphe connexe et $f_s : V \rightarrow \{0, 1\}$ la fonction caractéristique d'un ensemble $S \subseteq V$.

les bornes sur $\gamma_b(G)$

Proposition 2.3.1 [18] Soit G un graphe connexe, alors : $\gamma_b(G) \leq \min\{\text{Rad}(G), \gamma(G)\}$.

Théorème 2.3.2 [16] Soit G un graphe connexe, alors : $\gamma_b(G) \geq \left\lceil \frac{2\text{Rad}(G)}{3} \right\rceil$.

Théorème 2.3.3 [18] Soit G un graphe connexe non trivial, alors : $\gamma_b(G) \geq \left\lceil \frac{\text{Diam}(G)+1}{3} \right\rceil$.

De plus, pour tout graphe connexe G d'ordre n , $\left\lceil \frac{n}{3} \right\rceil$ est une borne supérieure de son nombre de broadcast domination.

Corollaire 2.3.1 [18] Pour tout graphe connexe G d'ordre n , alors : $\gamma_b(G) \leq \left\lceil \frac{n}{3} \right\rceil$.

Puisque $\text{Diam}(G) \geq \text{Rad}(G)$, une borne inférieure en terme de rayon est prouvée par Bresar et Spacapan pour le nombre de broadcast domination.

Proposition 2.3.2 [18] Soit G un graphe connexe. Si $\min\{\text{Rad}(G), \gamma(G)\} = k$, où $1 \leq k \leq 3$, alors : $\gamma_b(G) = k$.

2.4 Les variantes de la broadcast domination

Broadcast indépendant

Un broadcast f est dit *indépendant* si pour tout sommet $v \in V_f^+$, on a : $N_f[v] \cap V_f^+ = \{v\}$. Ou de manière équivalente, $|H(v)| = 1$.

Le coût maximum d'un broadcast indépendant de G , noté $\beta_0(G)$, est le nombre de broadcast d'indépendance.

Le nombre de broadcast d'indépendance inférieure, noté $i_b(G)$, est égal au coût minimum d'un broadcast indépendant maximal de G .

Pour tout graphe G , nous avons la chaîne d'inégalités suivante :

$$i_b(G) \leq \text{Rad}(G) \leq \text{Diam}(G) \leq \beta_b(G).$$

La fonction caractéristique f_S d'un ensemble indépendant maximal S dans un graphe G est un broadcast indépendant. On a donc : $i_b(G) \leq \beta_0(G) \leq \beta_b(G)$, mais f_S n'est pas nécessairement un broadcast indépendant maximal.

Broadcast dominant stable

Un broadcast f est qualifié de *dominant stable* s'il satisfait simultanément les conditions de domination et de stabilité. On désigne respectivement par $\Gamma_{ib}(G)$ et $\gamma_{ib}(G)$ le coût

maximum et minimum d'un broadcast dominant stable minimal sur G . Étant donné que la fonction caractéristique de tout ensemble stable maximal constitue un broadcast dominant stable minimal, nous en concluons que : $\gamma_{ib}(G) \leq i(G)$ et $\Gamma_{ib}(G) \geq \beta_0(G)$.

Broadcast efficace

[7] Un broadcast f est *efficace* si chaque sommet est f -dominé par un seul sommet broadcast. Le coût maximum d'un broadcast efficace est appelé le *nombre de broadcast efficace supérieur* et est noté $\Gamma_{eb}(G)$. Le nombre de broadcast efficace $\gamma_{eb}(G)$ est égal au coût minimum d'un broadcast efficace. À titre d'exemple, on peut dire que la fonction broadcast définie sur P_6 dans la figure 2.9 est un broadcast indépendant non efficace.

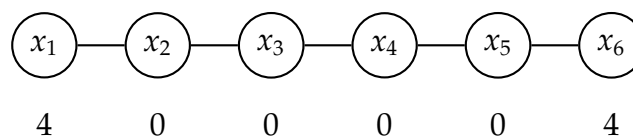


FIGURE 2.9 – Un broadcast stable non efficace sur le graphe P_6

Broadcast packing

[7] Un broadcast f est un *broadcast packing* si chaque sommet est f -dominé par au plus un sommet broadcast. Le coût maximum d'un broadcast packing de G est appelé le *nombre de broadcast packing*, et est noté $P_b(G)$. Le coût minimum d'un broadcast packing maximal est appelé le *nombre de broadcast packing inférieure* et est noté $p_b(G)$.

Notons que la fonction caractéristique f_S d'un packing maximal n'est pas forcément un broadcast packing maximal. Ceci peut être vu en considérant la chaîne P_5 . Soit c le centre de P_5 , le packing $S = \{c\}$ de figure 2.10 est maximal, mais f_S n'est pas un broadcast packing maximal, puisque la valeur $f_S(c) = 1$ peut être augmentée à 2 qui représente le broadcast packing maximal.

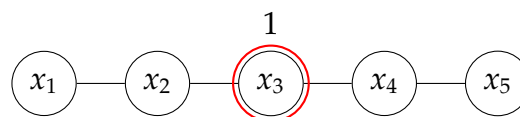


FIGURE 2.10 – Un packing maximal qui n'est pas un broadcast packing maximal

Broadcast domination sécurisée

Dans ce chapitre, nous étudions la notion de *Broadcast Domination Sécurisée dans les graphes*, en tant qu'extension des notions classiques de domination et de broadcast domination. Nous commençons par présenter les définitions et les notations de base nécessaires à la formalisation de ce concept, puis nous introduisons le paramètre qui représente le coût minimal d'une fonction de Broadcast Domination Sécurisée sur un graphe G . Ensuite, nous analysons ce paramètre en déterminant ses valeurs exactes pour certaines classes particulières de graphes, ainsi qu'en établissant des bornes inférieures et supérieures en fonction de propriétés structurelles du graphe, afin de fournir des estimations générales dans le cas général.

3.1 Définitions et notations

Dans cette section, nous présentons les définitions et les notations de base utilisées dans la suite de ce travail.

Définition 3.1.1 Soit $G = (V, E)$ un graphe connexe et $f : V \rightarrow \{0, 1, \dots, \text{diam}(G)\}$ une fonction broadcast, On dit que f est une fonction de broadcast domination sécurisée sur G si :

$$\forall u \in V \setminus V_f^+, \exists v \in V_f^+ \text{ tel que : } \begin{cases} u \in N_f(v), \\ (V_f^+ \setminus \{v\}) \cup \{u\} \text{ est un ensemble broadcast dominant.} \end{cases}$$

Le nombre de broadcast domination sécurisée de G , noté γ_{bs} , est défini par :

$$\gamma_{bs}(G) = \min \left\{ \sum_{v \in V} f(v) \mid f \text{ est une fonction de broadcast domination securisée sur } G \right\}.$$

3.2 Bornes sur γ_{bs}

Théorème 3.2.1 Soit G un graphe connexe, alors : $\gamma_{bs}(G) \leq \gamma_s(G)$.

Preuve. Soit $G = (V, E)$ un graphe et $V_f^+ \subseteq V$ un ensemble dominant sécurisé minimum, tel que : $\gamma_s(G) = |V_f^+|$.

Nous Construisons la fonction de broadcast f , On définit $f : V \rightarrow \{0, 1, \dots, \text{Diam}(G)\}$ comme suit : $f(v) = \begin{cases} 1 & \text{si } v \in V_f^+, \\ 0 & \text{sinon.} \end{cases}$

Maintenant, vérifions que la fonction f est une fonction de broadcast domination sécurisée, puisque V_f^+ est un ensemble dominant, chaque sommet $u \in V \setminus V_f^+$ est adjacent à au moins un sommet $v \in V_f^+$. Donc $d(u, v) = 1 = f(v)$.

Par hypothèse, V_f^+ est un ensemble dominant sécurisé. Pour tout $u \in V \setminus V_f^+$, il existe $v \in V_f^+ \cap N(u)$ tel que $V_f^{+'} = (V_f^+ \setminus \{v\}) \cup \{u\}$ est un ensemble dominant.

La fonction f est donc une fonction de broadcast sécurisée. Le coût total est : $\sum_{v \in V} f(v) = |V_f^+| = \gamma_s(G)$

Et comme $\gamma_{bs}(G)$ est le coût minimum parmi toutes les fonctions de broadcast sécurisées possibles sur G , alors : $\gamma_{bs}(G) \leq \gamma_s(G)$. $\square$

Théorème 3.2.2 (Cas d'un broadcast unique)

Soit G un graphe connexe d'ordre $n \geq 2$, alors : $\gamma_{bs}(G) \geq \left\lceil \frac{\text{Diam}(G)+1}{2} \right\rceil$.

Preuve. Soit $G = (V, E)$ un graphe et u, v deux sommets de G tel que $d(u, v) = \text{Diam}(G)$.

Pour que le sommet u soit dominé par une fonction de broadcast f , il doit exister un sommet w_1 tel que $f(w_1) \geq d(u, w_1)$. De même pour le sommet v , il doit exister w_2 tel que : $f(w_2) \geq d(v, w_2)$.

Si on utilise un seul sommet w pour dominer u et v simultanément, alors par l'inégalité triangulaire : $f(w) \geq d(u, w)$ et $f(w) \geq d(v, w)$, donc :

$$2f(w) \geq d(u, w) + d(v, w) \geq d(u, v), \text{ finalement : } f(w) \geq \frac{\text{Diam}(G)}{2}.$$

Le coût total $\sum_{v \in V} f(v)$ doit couvrir la distance maximale entre les sommets, ce qui impose la borne inférieure : $\gamma_{bs}(G) \geq \left\lceil \frac{\text{diam}(G)+1}{2} \right\rceil$.

□

Théorème 3.2.3 Soit G un graphe connexe d'ordre $n \geq 2$, alors : $\gamma_{bs}(G) \geq \gamma(G)$

Preuve. Soit $G = (V, E)$ un graphe et f une fonction de broadcast sécurisée minimale de coût $\gamma_{bs}(G)$.

Pour qu'un ensemble soit dominant, il suffit que chaque sommet $u \in V$ soit à une distance $d(u, v) \leq 1$ d'un sommet dominant v .

La fonction doit non seulement dominer tout le graphe, mais aussi satisfaire la condition de sécurité. Pour tout sommet $u \notin V_f^+$, il existe $v \in V_f^+$ qui, en le remplaçant, assure que $(V_f^+ \setminus \{v\}) \cup \{u\}$ est toujours dominant.

Comme la structure de broadcast sécurisé impose des contraintes supplémentaires et des portées de broadcast, le coût total $\sum_{v \in V} f(v)$ est supérieur ou égal à la domination simple $\gamma(G)$ sans sécurité.

□

Remarque 3.2.1 Bien que ces deux théorèmes fournissent des bornes inférieures différentes basées sur des propriétés distinctes du graphe (le diamètre et le nombre de domination simple), ils ne se contredisent pas. En effet, selon la structure du graphe G , l'une des deux bornes peut être plus restrictive (plus grande) que l'autre. Cela nous permet de combiner ces deux résultats dans le corollaire suivant pour obtenir une borne inférieure optimisée.

Corollaire 3.2.1 Pour tout graphe connexe G d'ordre $n \geq 2$, nous avons :

$$\gamma_{bs}(G) \geq \max \left(\left\lceil \frac{\text{Diam}(G) + 1}{2} \right\rceil, \gamma(G) \right).$$

Théorème 3.2.4 Soit G un graphe connexe d'ordre $n \geq 2$, alors : $\gamma_{bs}(G) \leq 2\gamma(G)$.

Preuve.

Soit $G = (V, E)$ un graphe et V_f^+ un ensemble dominant minimum de G , tel que : $\gamma(G) = |V_f^+|$. On définit $f : V \rightarrow \{0, 2\}$ comme suit :

$$f(v) = \begin{cases} 2 & \text{si } v \in V_f^+ \\ 0 & \text{sinon} \end{cases}$$

Puisque V_f^+ est un ensemble dominant, pour tout sommet $u \in V \setminus V_f^+$, il existe $v \in V_f^+$ tel que : $d(u, v) = 1 < f(v) = 2$, donc f est bien une fonction de broadcast dominante.

La puissance $f(v) = 2$ permet à chaque émetteur (sommet broadcast) de couvrir une large zone (voisins et voisins de voisins). Ce qui garantit une couverture totale du graphe même en cas de déplacement ou de changement de responsabilité pour assurer la sécurité.

Donc Le coût total est : $\sum_{v \in V} f(v) = \sum_{v \in S} 2 = 2|S| = 2\gamma(G)$

Puisque $\gamma_{bs}(G)$ est la valeur minimale parmi toutes les fonctions de broadcast sécurisées possibles, on en déduit : $\gamma_{bs}(G) \leq 2\gamma(G)$ $\square$

3.3 Broadcast domination sécurisée sur des classes de graphes

Dans cette section, nous allons étudier le paramètre γ_{bs} pour des classes particulières de graphes. On commence par les chaînes et les cycles.

Théorème 3.3.1 *Pour tout entier $n \geq 1$, $\gamma_{bs}(P_n) = \gamma_{bs}(C_n) = \left\lceil \frac{3n}{7} \right\rceil$.*

Preuve.

Soit f une fonction de broadcast sécurisée sur la chaîne P_n . Nous commençons à prouver que $\gamma_{bs} \geq \left\lceil \frac{3n}{7} \right\rceil$.

Soit P_n une chaîne dont les sommets sont $v_1, v_2, \dots, v_n$. Pour tout $i \in \{1, \dots, n-6\}$, on définit le bloc de sept sommets consécutifs : $B_i = \{v_i, v_{i+1}, v_{i+2}, v_{i+3}, v_{i+4}, v_{i+5}, v_{i+6}\}$, dans la preuve, on note simplement B un tel bloc quelconque.

Considérons toute sous-chaîne de 7 sommets consécutifs.

Nous affirmons que : $\sum_{v \in B} f(v) \geq 3$.

En effet, si $\sum f(v) \leq 2$, alors les cas possibles sont que :

- un sommet de valeur 2, qui ne peut couvrir au plus que 5 sommets,

- deux sommets de valeur 1, couvrant au plus 6 sommets.

Dans tous les cas, on ne peut pas couvrir 7 sommets, ce qui contredit le fait que f est une fonction dominante. Ainsi, chaque bloc de 7 sommets nécessite au moins 3 unités de poids, ce qui donne une densité minimale de $\frac{3}{7}$. Par conséquent, $\gamma_{bs} \geq \frac{3n}{7}$. Comme γ_{bs} est entier, on obtient : $\gamma_{bs} \geq \left\lceil \frac{3n}{7} \right\rceil$. Donc, $\gamma_{bs}(P_n) \geq \left\lceil \frac{3n}{7} \right\rceil$.

Maintenant nous prouvons que : $\gamma_{bs}(P_n) \geq \left\lceil \frac{3n}{7} \right\rceil$.

D'après les informations précédentes, nous avons l'inégalité générale suivante pour tout graphe G : $\gamma_{bs}(G) \leq \gamma_s(G)$, donc : $\gamma_{bs}(P_n) \leq \gamma_s(P_n) = \left\lceil \frac{3n}{7} \right\rceil$.

□

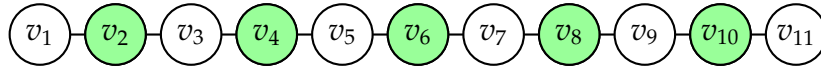


FIGURE 3.1 – Chaîne P_{11} avec un broadcast sécurisé : $\gamma_{bs}(P_{11}) = 5$

Proposition 3.3.1 Soit $K_{n,m}$ un graphe biparti complet avec $n \leq m$, alors : $\gamma_{bs}(K_{n,m}) = 2$.

preuve Supposons, par contradiction, qu'il existe une fonction de domination broadcast sécurisée f de poids 1. Alors il existe un unique sommet $v \in V$ tel que $f(v) = 1$ et $f(u) = 0$ pour tout $u \neq v$.

Soit (A, B) la bipartition de $K_{n,m}$ et supposons $v \in A$, alors v domine tous les sommets de B , mais aucun sommet de $A \setminus \{v\}$ n'est à distance ≤ 1 de v . En effet, la distance entre deux sommets de A est égale à 2.

Ainsi, ces sommets ne sont pas dominés, ce qui contredit le fait que f est une fonction de domination broadcast. Le même raisonnement s'applique si $v \in B$. Donc $\gamma_{bs}(K_{n,m}) \neq 1$.

Construisons une fonction de poids 2. Choisissons un sommet $v \in A$ et définissons $f : V \rightarrow \{0, 1, 2\}$ par : $f(v) = 2$ et $f(u) = 0$ pour tout $u \neq v$.

- Tout sommet $u \in B$ est adjacent à v (car $K_{n,m}$ est biparti complet), donc $d(u, v) = 1 \leq 2$: u est dominé.
- Le sommet v lui-même est dominé.
- Pour tout autre sommet $v' \in A \setminus \{v\}$, la chaîne $v' - u - v$ (où u est n'importe quel sommet de B) a longueur 2, donc $d(v, v') = 2 \leq 2$: v' est dominé.
- **Si** $v \in A$: Alors v est adjacent à tous les sommets de B (distance 1), et pour tout autre sommet $v' \in A \setminus \{v\}$, la distance de v à v' est 2.
- **Si** $v \in B$: Alors v est adjacent à tous les sommets de A (distance 1), et pour tout autre sommet $v' \in B \setminus \{v\}$, la distance de v à v' est 2.

Dans les deux cas, la domination est conservée. La condition de sécurité est donc satisfaite.

Par conséquent, f est une fonction de domination broadcast sécurisée de poids 2, ce qui prouve $\gamma_{bs}(K_{n,m}) = 2$.

Observation 1 Dans le graphe complet K_n , on a : $\gamma_{bs}(K_n) = 1$. Un seul sommet actif suffit pour assurer une domination broadcast sécurisée de l'ensemble du graphe.

Amélioration et sécurisation de l'ensemble broadcast dominant

Dans le chapitre précédent, nous avons introduit le paramètre **Broadcast domination sécurisée**, noté $\gamma_{bs}(G)$, et analysé son comportement sur certaines classes particulières de graphes, telles que les chaînes, les cycles et les graphes complets. Nous avons obtenu des résultats exacts ainsi que des bornes théoriques, ce qui a permis de mieux appréhender les propriétés de ce paramètre au sein de structures bien définies.

Cependant, dans le cas général, la détermination exacte de $\gamma_{bs}(G)$ pour un graphe quelconque demeure un problème difficile, souvent associé à des questions complexes d'optimisation combinatoire. En effet, à l'instar de nombreux paramètres de domination, son calcul peut s'avérer très coûteux en temps, voire irréalisable pour des graphes de grande taille.

Pour surmonter cette difficulté, il est pertinent de recourir à des méthodes approchées, en particulier à des algorithmes heuristiques, qui offrent la possibilité d'obtenir des solutions proches de l'optimal en un temps raisonnable.

Dans ce chapitre, nous présentons une approche heuristique visant à estimer la valeur de $\gamma_{bs}(G)$ pour un graphe général. Cette méthode s'appuie sur l'algorithme de Lokshtanov[28], reconnu pour son efficacité dans la résolution de problèmes de domination dans les graphes.

Nous commencerons par exposer le principe général de cette approche, puis nous détaillerons les algorithmes employés. Enfin, nous évaluerons leurs performances à l'aide d'expérimentations menées sur des instances de test.

4.1 L'heuristique proposée

L'heuristique que nous allons proposer s'articule autour de trois phases principales :

- **Phase 1** : Détermination d'un ensemble de broadcast domination en s'appuyant sur l'algorithme de Lokshtanov.
- **Phase 2** : Identification et sécurisation des adjacents privés des sources de l'ensemble de broadcast domination.
- **Phase 3** : Sécurisation des sommets à couverture multiple.

Détermination d'un ensemble broadcast dominant

Cette première étape consiste à trouver un ensemble de sommets (sources) tel que chaque sommet du graphe soit couvert, directement ou indirectement, par un signal de broadcast émis par ces sources. Pour ce faire, nous nous appuyons sur l'algorithme de Lokshtanov, qui permet de construire efficacement un ensemble de broadcast domination de taille minimale.

Procédure de l'algorithme de Lokshtanov

L'algorithme de Lokshtanov permet de déterminer un ensemble de broadcast domination de coût minimum en construisant, pour chaque sommet $u \in V$, un graphe orienté auxiliaire G_u .

- **Choix du sommet de référence**

On fixe un sommet $u \in V$ qui représente l'extrémité gauche du chemin que l'on souhaite construire.

- **Construction des sommets de G_u**

Pour tout $v \in V$ et tout $p \in \{1, \dots, \text{Rad}(G)\}$, on crée un sommet (v, p) dans G_u si et seulement si l'une des conditions suivantes est vérifiée :

- * $G(V \setminus B(v, p))$ est connexe (ou vide) et $u \in B(v, p)$,
- * $G(V \setminus B(v, p))$ possède au plus deux composantes connexes et $u \notin B(v, p)$.

Chaque sommet (v, p) représente une affectation $f(v) = p$ et possède un poids égal à p .

- **Définition des composantes**

Pour chaque sommet (v, p) , on considère :

- * $L_u(v, p)$: la composante connexe de $G(V \setminus B(v, p))$ contenant u (partie gauche),
- * $R_u(v, p)$: la composante connexe ne contenant pas u (partie droite).

On a alors : $L_u(v, p) = \emptyset$ pour $(v, p) \in A_u \cup D_u$, $R_u(v, p) = \emptyset$ pour $(v, p) \in C_u \cup D_u$.

- **Partition des sommets**

Les sommets de G_u sont répartis en quatre ensembles :

- * $A_u = \{(v, p) \mid G(V \setminus B(v, p)) \text{ est connexe et } u \in B(v, p)\},$
- * $B_u = \{(v, p) \mid G(V \setminus B(v, p)) \text{ possède deux composantes connexes}\},$
- * $C_u = \{(v, p) \mid G(V \setminus B(v, p)) \text{ est connexe et } u \notin B(v, p)\},$
- * $D_u = \{(v, p) \mid B(v, p) = V\}.$

- **Construction des arêtes de G_u**

Une arête orientée $(v, p) \rightarrow (w, q)$ est ajoutée si et seulement si les conditions suivantes sont satisfaites :

- * $B(v, p) \cap B(w, q) = \emptyset,$
- * $R_u(v, p) \neq \emptyset$ et $L_u(w, q) \neq \emptyset,$
- * $(N_G(B(w, q)) \cap L_u(w, q)) \subseteq B(v, p)$ et $(N_G(B(v, p)) \cap R_u(v, p)) \subseteq B(w, q).$

- **Calcul du coût**

Le coût d'un chemin est la somme des poids des sommets qui le composent :

$$\sum_{(v,p) \in \text{chemin}} p$$

- **Choix de la solution approchée**

On répète le processus pour tout $u \in V$ et on sélectionne le broadcast de coût minimal.

Algorithme de Lokshtanov (MPBD)

Algorithm 1 Domination par broadcast du plus court chemin MPBD

Entrée : Un graphe $G = (V, E)$

Sortie : Une fonction de broadcast domination efficace f sur G telle que $c_f(V) \leq \gamma_{bp}(G)$

```

1: Pour chaque sommet  $v \in V$  faire
2:    $f(v) \leftarrow 0$ 
3:   Soit  $P$  un chemin fictif avec  $W(P) = \text{rad}(G) + 1$ 
4:   Pour chaque sommet  $u \in V$  faire
5:     Construire le graphe  $G_u$  avec les ensembles  $A_u, B_u, C_u, D_u$ 
6:     Trouver un chemin de poids minimum  $P_u$  commençant dans un sommet de
7:      $A_u \cup D_u$  et se terminant dans un sommet de  $C_u \cup D_u$ 
8:     Si  $W(P_u) < W(P)$  alors
9:        $P \leftarrow P_u$ 
10:    Fin Si
11:  Fin Pour
12: Fin Pour
13: Pour chaque sommet  $(v, p) \in P$  faire
14:    $f(v) \leftarrow p$ 
15: Fin Pour

```

Déroulement de l'algorithme

Dans ce qui suit, nous déroulons l'algorithme sur un exemple typique afin d'illustrer concrètement chacune de ses étapes et de mettre en évidence son fonctionnement pratique.

Exemple 4.1.1 Soit $G(V, E)$ un graphe tel que : $|V| = 7$, $|E| = 8$ et $\text{Rad}(G) = 3$.

Nous construisons un graphe auxiliaire G_u pour chaque $u \in V$.

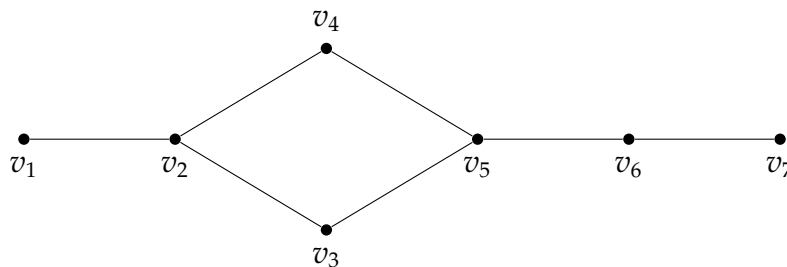


FIGURE 4.1 – Application de l'algorithme lokshtanov

Pour tout $v \in V : f(v_1) = f(v_2) = f(v_3) = f(v_4) = f(v_5) = f(v_6) = f(v_7) = 0$;

Soit P une chaîne fictive $W(P)=3+1=4$;

Itération 1 : soit $u = v_1$, pour tout $v \in V$, pour chaque $p \in \{1, 2, 3\}$.

On a : $V(G_u) = \{(v_1, 1), (v_1, 2), (v_1, 3), (v_2, 1), (v_2, 2), (v_2, 3), (v_3, 2), (v_3, 3), (v_4, 2), (v_4, 3), (v_5, 1), (v_5, 2), (v_5, 3), (v_6, 1), (v_6, 2), (v_6, 3), (v_7, 1), (v_7, 2), (v_7, 3)\}$.

Mais $\{(v_3, 1), (v_4, 1)\}$ ne sont pas dans $V(G_u)$.

Donc $|G_u| = 19 \leq 21$. On trouve la chaîne P_u qui commence par $A_u \cup D_u$ et se termine par $C_u \cup D_u$.

D'abord on construit A_u, B_u, C_u, D_u :

$A_u = \{(v_1, 1), (v_1, 2), (v_1, 3), (v_2, 1), (v_2, 2), (v_2, 3), (v_3, 2), (v_4, 2)\}$

$B_u = \{(v_5, 1)\}$

$C_u = \{(v_5, 2), (v_6, 1), (v_6, 2), (v_6, 3), (v_7, 1), (v_7, 2), (v_7, 3)\}$

$D_u = \{(v_3, 3), (v_4, 3), (v_5, 3)\}$

Maintenant, on donne les arêtes de G_u :

$(v_1, 1) \rightarrow (v_5, 1), (v_1, 1) \rightarrow (v_6, 2), (v_1, 1) \rightarrow (v_7, 3), (v_1, 2) \rightarrow (v_6, 1), (v_1, 2) \rightarrow (v_7, 2), (v_1, 3) \rightarrow (v_7, 1), (v_2, 1) \rightarrow (v_6, 1), (v_2, 1) \rightarrow (v_7, 2), (v_2, 2) \rightarrow (v_7, 1)$.

Soit : $pu_1 = (v_1, 1) \rightarrow (v_6, 2), pu_2 = (v_1, 1) \rightarrow (v_7, 3), pu_3 = (v_1, 2) \rightarrow (v_6, 1), pu_4 = (v_1, 2) \rightarrow (v_7, 2), pu_5 = (v_1, 3) \rightarrow (v_7, 1), pu_6 = (v_2, 1) \rightarrow (v_6, 1), pu_7 = (v_2, 1) \rightarrow (v_7, 2), pu_8 = (v_2, 2) \rightarrow (v_7, 1)$.

Donc : la chaîne $P_{v_1} = pu_6$, avec : $W(P_u) = 2 < 4$.

Itération 2 : soit $u = v_2$, pour tout $v \in V$ pour chaque $p \in \{1, 2, 3\}$

on a : $V(G_u) = \{(v_1, 1), (v_1, 2), (v_1, 3), (v_2, 1), (v_2, 2), (v_2, 3), (v_3, 2), (v_3, 3), (v_4, 2), (v_4, 3), (v_5, 1), (v_5, 2), (v_5, 3), (v_6, 1), (v_6, 2), (v_6, 3), (v_7, 1), (v_7, 2), (v_7, 3)\}$.

Mais $\{(v_3, 1), (v_4, 1)\}$ ne sont pas de $V(G_u)$.

Donc $|G_u| = 19 \leq 21$, On trouve la chaîne P_u commence par $A_u \cup D_u$ et se termine par $C_u \cup D_u$.

D'abord on construire A_u, B_u, C_u, D_u .

$A_u = \{(v_1, 1), (v_1, 2), (v_1, 3), (v_2, 1), (v_2, 2), (v_2, 3), (v_3, 2), (v_4, 2), (v_5, 2), (v_6, 3)\}$

$B_u = \{(v_5, 1)\}$

$C_u = \{(v_6, 1), (v_6, 2), (v_7, 1), (v_7, 2), (v_7, 3)\}$

$D_u = \{(v_3, 3), (v_4, 3), (v_5, 3)\}$

Maintenant, on donne les arête de G_u :

$(v_1, 1) \rightarrow (v_5, 1), (v_1, 1) \rightarrow (v_6, 2), (v_1, 1) \rightarrow (v_7, 3), (v_1, 2) \rightarrow (v_6, 1), (v_1, 2) \rightarrow (v_7, 2), (v_1, 3) \rightarrow (v_7, 1), (v_2, 1) \rightarrow (v_6, 1), (v_2, 1) \rightarrow (v_7, 2), (v_2, 2) \rightarrow (v_7, 1)$

soit : $pu_1 = (v_1, 1) \rightarrow (v_6, 2)$, $pu_2 = (v_1, 1) \rightarrow (v_7, 3)$, $pu_3 = (v_1, 2) \rightarrow (v_6, 1)$, $pu_4 = (v_1, 2) \rightarrow (v_7, 2)$, $pu_5 = (v_1, 3) \rightarrow (v_7, 1)$, $pu_6 = (v_2, 1) \rightarrow (v_6, 1)$, $pu_7 = (v_2, 1) \rightarrow (v_7, 2)$, $pu_8 = (v_2, 2) \rightarrow (v_7, 1)$

Donc : la chaîne $pv_2 = pu_6$ avec $W(P_u) = 2 < 4$.

Itération 3 : soit $u = v_3$, pour tout $v \in V$ pour chaque $p \in \{1, 2, 3\}$.

on a : $V(G_u) = \{(v_1, 1), (v_1, 2), (v_1, 3), (v_2, 1), (v_2, 2), (v_2, 3), (v_3, 2), (v_3, 3), (v_4, 2), (v_4, 3), (v_5, 2), (v_5, 3), (v_6, 1), (v_6, 2), (v_6, 3), (v_7, 1), (v_7, 2), (v_7, 3)\}$

Mais $\{(v_3, 1), (v_4, 1), (v_5, 1)\}$ ne sont pas de $V(G_u)$.

Donc : $|G_u| = 18 \leq 21$, On trouve la chaîne P_u commence par $A_u \cup D_u$ et se termine par $C_u \cup D_u$.

D'abord on construire A_u, B_u, C_u, D_u .

$A_u = \{(v_1, 2), (v_1, 3), (v_2, 1), (v_2, 2), (v_2, 3), (v_3, 2), (v_4, 2), (v_5, 2), (v_6, 2), (v_6, 3), (v_7, 3)\}$

$B_u = \emptyset$

$C_u = \{(v_1, 1), (v_6, 1), (v_7, 1), (v_7, 2)\}$

$D_u = \{(v_3, 3), (v_4, 3), (v_5, 3)\}$

Maintenant, on donne les arête de G_u :

$(v_1, 2) \rightarrow (v_6, 1), (v_1, 2) \rightarrow (v_7, 2), (v_1, 3) \rightarrow (v_7, 1), (v_2, 1) \rightarrow (v_6, 1), (v_2, 1) \rightarrow (v_7, 2), (v_2, 2) \rightarrow (v_7, 1)$

soit : $pu_1 = (v_1, 2) \rightarrow (v_6, 1)$, $pu_2 = (v_1, 2) \rightarrow (v_7, 2)$, $pu_3 = (v_1, 3) \rightarrow (v_7, 1)$, $pu_4 = (v_2, 1) \rightarrow (v_6, 1)$, $pu_5 = (v_2, 1) \rightarrow (v_7, 2)$, $pu_6 = (v_2, 2) \rightarrow (v_7, 1)$

donc : la chaîne $pv_3 = pu_4$ avec $W(P_u) = 2 < 4$.

Itération 4 : soit $u = v_4$, pour tout $v \in V$ pour chaque $p \in \{1, 2, 3\}$.

on a : $V(G_u) = \{(v_1, 1), (v_1, 2), (v_1, 3), (v_2, 1), (v_2, 2), (v_2, 3), (v_3, 2), (v_3, 3), (v_4, 2), (v_4, 3), (v_5, 2), (v_5, 3), (v_6, 1), (v_6, 2), (v_6, 3), (v_7, 1), (v_7, 2), (v_7, 3)\}$

Mais $\{(v_3, 1), (v_4, 1), (v_5, 1)\}$ ne sont pas de $V(G_u)$.

Donc : $|G_u| = 18 \leq 21$, On trouve la chaîne P_u commence par $A_u \cup D_u$ et se termine par $C_u \cup D_u$.

D'abord on construire A_u, B_u, C_u, D_u .

$A_u = \{(v_1, 2), (v_1, 3), (v_2, 1), (v_2, 2), (v_2, 3), (v_3, 2), (v_4, 2), (v_5, 2), (v_6, 2), (v_6, 3), (v_7, 3)\}$

$B_u = \emptyset$

$C_u = \{(v_1, 1), (v_6, 1), (v_7, 1), (v_7, 2)\}$

$D_u = \{(v_3, 3), (v_4, 3), (v_5, 3)\}$

Maintenant, on donne les arête de G_u :

$(v_1, 2) \rightarrow (v_6, 1), (v_1, 2) \rightarrow (v_7, 2), (v_1, 3) \rightarrow (v_7, 1), (v_2, 1) \rightarrow (v_6, 1), (v_2, 1) \rightarrow (v_7, 2), (v_2, 2) \rightarrow (v_7, 1)$

soit : $pu_1 = (v_1, 2) \rightarrow (v_6, 1)$, $pu_2 = (v_1, 2) \rightarrow (v_7, 2)$, $pu_3 = (v_1, 3) \rightarrow (v_7, 1)$, $pu_4 = (v_2, 1) \rightarrow (v_6, 1)$, $pu_5 = (v_2, 1) \rightarrow (v_7, 2)$, $pu_6 = (v_2, 2) \rightarrow (v_7, 1)$

donc : la chaine $pv_4 = pu_4$ avec $W(P_u) = 2 < 4$

Itération 5 : soit $u = v_5$, pour tout $v \in V$ pour chaque $p \in \{1, 2, 3\}$.

on a : $V(G_u) = \{(v_1, 1), (v_1, 2), (v_1, 3), (v_2, 1), (v_2, 2), (v_2, 3), (v_3, 2), (v_3, 3), (v_4, 2), (v_4, 3), (v_5, 2), (v_5, 3), (v_6, 1), (v_6, 2), (v_6, 3), (v_7, 1), (v_7, 2), (v_7, 3)\}$

Mais $\{(v_4, 1), (v_3, 1), (v_5, 1)\}$ ne sont pas de $V(G_u)$.

Donc : $|G_u| = 18 \leq 21$, On trouve la chaine P_u commence par $A_u \cup D_u$ et se termine par $C_u \cup D_u$

D'abord on construire A_u, B_u, C_u, D_u .

$A_u = \{(v_1, 3), (v_2, 2), (v_2, 3), (v_3, 2), (v_4, 2), (v_5, 2), (v_6, 1), (v_6, 2), (v_6, 3), (v_7, 2), (v_7, 3)\}$

$B_u = \emptyset$

$C_u = \{(v_1, 1), (v_1, 2), (v_2, 1), (v_7, 1)\}$

$D_u = \{(v_3, 3), (v_4, 3), (v_5, 3)\}$

Maintenant, on donne les arête de G_u :

$(v_1, 3) \rightarrow (v_7, 1), (v_2, 2) \rightarrow (v_7, 1)$

soit : $pu_1 = (v_1, 3) \rightarrow (v_7, 1)$, $pu_2 = (v_2, 2) \rightarrow (v_7, 1)$

donc : la chaine $pu = pu_2$ avec le poids minimum 2.

Itération 6 : soit $u = v_6$, pour tout $v \in V$ pour chaque $p \in \{1, 2, 3\}$.

on a : $V(G_u) = \{(v_1, 1), (v_1, 2), (v_1, 3), (v_2, 1), (v_2, 3), (v_3, 2), (v_3, 3), (v_4, 2), (v_4, 3), (v_5, 2), (v_5, 3), (v_6, 1), (v_6, 2), (v_6, 3), (v_7, 1), (v_7, 2), (v_7, 3)\}$

Mais $\{(v_3, 1), (v_4, 1), (v_5, 1)\}$ ne sont pas de $V(G_u)$.

Donc : $|G_u| = 17 \leq 21$, On trouve la chaine P_u commence par $A_u \cup D_u$ et se termine par $C_u \cup D_u$

D'abord on construire A_u, B_u, C_u, D_u .

$A_u = \{(v_2, 3), (v_3, 2), (v_4, 2), (v_5, 2), (v_6, 1), (v_6, 2), (v_6, 3), (v_7, 1), (v_7, 2), (v_7, 3)\}$

$B_u = \emptyset$

$D_u = \{(v_3, 3), (v_4, 3), (v_5, 3)\}$

Maintenant, on donne les arête de G_u :

$E(G_u) = \emptyset$

Itération 7 : soit $u = v_7$, pour tout $v \in V$ pour chaque $p \in \{1, 2, 3\}$.

on a : $V(G_u) = \{(v_1, 1), (v_1, 2), (v_1, 3), (v_2, 1), (v_2, 2), (v_2, 3), (v_3, 2), (v_3, 3), (v_4, 2), (v_4, 3), (v_5, 1), (v_5, 2), (v_5, 3), (v_6, 1), (v_6, 2), (v_6, 3), (v_7, 1), (v_7, 2), (v_7, 3)\}$.

Mais $\{(v_3, 1), (v_4, 1)\}$ ne sont pas de $V(G_u)$.

Donc : $|G_u| = 19 \leq 21$, On trouve la chaîne P_u commence par $A_u \cup D_u$ et se termine par $C_u \cup D_u$

D'abord on construit A_u, B_u, C_u, D_u .

$$A_u = \{(v_5, 2), (v_6, 1), (v_6, 2), (v_6, 3), (v_7, 1), (v_7, 2), (v_7, 3)\}$$

$$B_u = \{(v_5, 1)\}$$

$$C_u = \{(v_1, 1), (v_1, 2), (v_1, 3), (v_2, 1), (v_2, 2), (v_2, 3), (v_3, 2), (v_4, 2)\}$$

$$D_u = \{(v_3, 3), (v_4, 3), (v_5, 3)\}$$

Maintenant, on donne les arête de G_u :

$$E(G_u) = \emptyset$$

Après les sept itérations, pour tout $u \in G$ le plus courte chemine permet les P_u est $P = (v_2, 1) \rightarrow (v_6, 1)$ alors $f(v_2) = 1; f(v_6) = 1$

$$\text{donc : } c_f(V) = f(v_2) + f(v_6) = 1 + 1 = 2.$$

Sécurisation des adjacents privés des sources de broadcast

Après avoir déterminé un ensemble de broadcast dominant de coût minimal, nous cherchons à l'adapter pour obtenir un ensemble de broadcast dominant sécurisé. Pour ce faire, nous commençons par analyser l'ensemble de broadcast dominant V_f^+ précédemment défini.

Dans cet ensemble, on distingue deux sous-ensembles parmi les sommets dominés :

- $P_{V_f^+}$, l'ensemble des adjacents privés, c'est-à-dire les sommets dominés par un seul sommet de V_f^+ ;
- l'ensemble des autres sommets, dominés par au moins deux sommets de V_f^+ .

Si l'ensemble $P_{V_f^+}$ est vide, alors V_f^+ constitue un ensemble de broadcast dominant sécurisé. Cette observation conduit à l'énoncé du lemme suivant :

Lemme 4.1.1 *Pour tout graphe G , on a : $\gamma_{bs}(G) \leq \gamma_{2b}(G)$.*

Preuve. Supposons par l'absurde que $\gamma_{bs}(G) > \gamma_{2b}(G)$.

Soit B_2 un ensemble $2b$ -dominant de taille minimum. Supposons qu'il existe un sommet v qui n'est pas sécurisé par B_2 .

Alors il existe $t \in B_2$ et $u \notin B_2$ tels que v ne soit pas dominé par $(B_2 \setminus \{t\}) \cup \{u\}$.

Cela signifie que v est dominé uniquement par t , ce qui contredit la définition d'un ensemble $2b$ -dominant, où chaque sommet dominé l'est par au moins deux éléments de B_2 .

La supposition de départ est donc fausse, d'où le résultat. $\square$

Sur la base de ce résultat, nous adaptons l'ensemble de broadcast dominant initial afin de déterminer un ensemble de broadcast dominant sécurisé. Pour cela, nous définissons, pour chaque source $x \in V_f^+$, les sommets couverts uniquement par cette source, appelés adjacents privés de x :

$$\text{priv}(x) = \{u \in V \setminus V_f^+ : d(x, u) \leq f(x) \text{ et } \forall y \in V_f^+ \setminus \{x\}, d(y, u) > f(y)\}.$$

Ensuite, pour sécuriser ces adjacents privés, on calcule la distance (plus court chemin) entre chaque paire de sommets de cet ensemble. On identifie alors les deux sommets dont la distance est maximale. Le broadcast de la source x est alors ajusté pour correspondre au maximum entre son broadcast initial et cette distance maximale. Cette formulation met en avant la logique de l'ajustement du broadcast afin de garantir la sécurité des adjacents privés.

l'algorithme de la phase 2

Cette phase consiste à sécuriser les sommets privés associés à chaque source $s \in V_f^+$. Elle commence par l'identification des adjacents privés, c'est-à-dire les sommets couverts uniquement par s . Ensuite, les distances entre ces sommets sont calculées afin de déterminer la distance maximale. Cette valeur est ensuite comparée au broadcast de la source. Si elle est plus grande, le broadcast est ajusté, sinon il reste inchangé. L'opération est répétée pour toutes les sources de V_f^+ .

Algorithm 2 Phase 2 : Étape de sécurisation des sommets privés

Entrée : $G = (V, E)$, $V_f^+ = \{(s_1, f_1(s_1)), \dots, (s_t, f_t(s_t))\}$

Sortie : $V_f^+ = \{(s_1, f'_1(s_1)), \dots, (s_t, f'_t(s_t))\}$

```

1: Pour chaque  $s \in V_f^+$  faire
2:    $P(s) \leftarrow \emptyset$ 
3:   Pour chaque  $v \in V \setminus \{s\}$  faire
4:     Si  $d(v, s) \leq f(s)$  alors
5:        $P(s) \leftarrow P(s) \cup \{v\}$ 
6:     Fin Si
7:   Fin Pour
8:   Pour chaque  $v \in P(s)$  faire
9:     Pour chaque  $s' \in V_f^+ \setminus \{s\}$  faire
10:      Si  $d(v, s') \leq f(s')$  alors
11:         $P(s) \leftarrow P(s) \setminus \{v\}$ 
12:      Fin Si
13:    Fin Pour
14:  Fin Pour
15:   $d \leftarrow 0$ 
16:  Pour chaque  $v \in P(s)$  faire
17:    Pour chaque  $u \in P(s)$  faire
18:      Si  $d(v, u) > d$  alors
19:         $d \leftarrow d(v, u)$ 
20:      Fin Si
21:    Fin Pour
22:  Fin Pour
23:  Si  $d > f(s)$  alors
24:     $f'(s) \leftarrow d$ 
25:  else
26:     $f'(s) \leftarrow f(s)$ 
27:  Fin Si
28:   $f(s) \leftarrow f'(s)$ 
29: Fin Pour

```

Remarque 4.1.1 L'algorithme est dynamique : à chaque étape, il recalcule les adjacents privés des sommets concernés, en tenant compte des modifications apportées aux valeurs de broadcast.

Déroulement de l'algorithme

Cette section illustre, à travers un exemple, les différentes étapes de l'algorithme en détaillant le processus de sécurisation de l'ensemble de broadcast domination, notamment par l'ajustement dynamique des valeurs de broadcast et la gestion des adjacents privés.

Exemple 4.1.2 Dans cet exemple, nous allons exécuter la phase 2 sur le graphe représenté dans la Figure 4.1. L'exécution de l'algorithme de Lokshtanov sur ce graphe donne initialement : $V_f^+ = \{v_2, v_6\}$, $f(v_2) = f(v_6) = 1$.

Étape 1 :

On détermine les adjacents privés de v_2 : $P(v_2) = \{v_1, v_3, v_4\}$.

On calcule les distances entre les éléments de $P(v_2)$:

$$d(v_1, v_3) = 2, d(v_1, v_4) = 2, d(v_3, v_4) = 2.$$

On choisit la plus grande distance parmi les distances calculées, donc : $d = 2$.

Comme $d > f(v_2)$ ($2 > 1$), on met à jour le broadcast : $2 > 1 \Rightarrow f'(v_2) = 2$ et $f(v_2) = 2$.

Étape 2 :

On détermine les adjacents privés de v_6 : $P(v_6) = \{v_7\}$. Comme il y a un seul sommet adjacent, la valeur ne change pas : $f'(v_6) = 1$.

Résultat final de la phase 2 :

$$f'(v_2) = 2, \quad f'(v_6) = 1.$$

$$P(v_2) = \{v_1, v_3, v_4\}, \quad P(v_6) = \{v_7\}.$$

Sécurisation des sommets à couverture multiple

Cette phase consiste à sécuriser les sommets couverts par plusieurs sources, en définissant d'abord le **défenseur**.

Le défenseur est la source qui se déplace virtuellement et qui s'occupe de la sécurisation des sommets. Ce défenseur est défini comme la source qui **minimise l'écart absolu** entre son broadcast actuel et le broadcast requis pour maintenir la domination, par rapport aux autres sources candidates : $\text{def}(u) = \arg \min_{s \in \text{Couvert}(u)} |f'(s) - dm(s)|$

Où $dm(s) = \max_{v \in \text{Priv}(s)} d(u, v)$ représente le broadcast nécessaire pour couvrir les adjacents privés depuis la nouvelle position u .

Une fois le défenseur déterminé, on met à jour son broadcast. On considère :

$$d = \max_{v \in \text{Priv}(\text{def}(u))} d(u, v)$$

Si $f'(\text{def}(u)) < d$, alors on pose : $f''(\text{def}(u)) = d$ Sinon, le broadcast reste inchangé : $f''(\text{def}(u)) = f'(\text{def}(u))$.

Ainsi, $f''(\text{def}(u))$ représente le nouveau broadcast du défenseur garantissant la sécurisation du sommet u .

l'algorithme de la phase 3

Algorithm 3 Phase 3 : Étape de sécurisation des sommets à couverture multiple

Entrée : $G = (V, E)$, $V_f^{+'} = \{(s_1, f'_1(s_1)), \dots, (s_t, f'_t(s_t))\}$

Sortie : $V_f^{+''} = \{(s_1, f''_1(s_1)), \dots, (s_t, f''_t(s_t))\}$

```

1: Pour chaque sommet  $u \in V \setminus (V_f^{+'} \cup \text{Priv}V_f^{+'})$  faire
2:    $\text{Couvert}(u) \leftarrow \emptyset$ 
3:   Pour chaque source  $s_i \in (V_f^{+'})$  faire
4:     Si  $d(s_i, u) \leq f'_i(s_i)$  alors
5:        $\text{Couvert}(u) \leftarrow \text{Couvert}(u) \cup \{s_i\}$ 
6:     Fin Si
7:   Fin Pour
8:   Si  $\exists s \in \text{Couvert}(u)$  tel que  $\text{Priv}(s) = \emptyset$  alors
9:      $\text{defenseur} \leftarrow s$ 
10:     $f''(\text{defenseur}) \leftarrow f'(\text{defenseur})$ 
11:   else
12:     Pour chaque  $s \in \text{Couvert}(u)$  faire
13:        $dm(s) \leftarrow \max_{v \in \text{Priv}(s)} d(u, v)$ 
14:     Fin Pour
15:      $\text{defenseur} \leftarrow \arg \min_{s \in \text{Couvert}(u)} |f'(s) - dm(s)|$ 
16:     Si  $f'(\text{defenseur}) < dm(\text{defenseur})$  alors
17:        $f''(\text{defenseur}) \leftarrow dm(\text{defenseur})$ 
18:     else
19:        $f''(\text{defenseur}) \leftarrow f'(\text{defenseur})$ 
20:     Fin Si
21:   Fin Si
22:    $f'(\text{defenseur}) \leftarrow f''(\text{defenseur})$ 
23:    $f(\text{defenseur}) \leftarrow f''(\text{defenseur})$ 
24: Fin Pour
```

le déroulement d'algorithme

Cette section illustre, à travers un exemple, les différentes étapes de l'algorithme en détaillant le processus de sécurisation des sommets à couverture multiple. L'objectif est d'identifier les sommets couverts par plusieurs sources et de sélectionner, pour chacun d'eux, un défenseur approprié afin de garantir le maintien de la couverture et de renforcer la propriété de broadcast domination sécurisée.

Exemple 4.1.3 Après la Phase 2, $V_f^{+'} = \{(v_2, 2), (v_6, 1)\}$.

Le sommet identifié par $u \in V \setminus (V_f^{+'} \cup \text{Priv}(V_f^{+'}))$, ce qui donne $u = v_5$.

Identification des sources candidates :

- $d(v_2, v_5) = 2 \leq f'(v_2) \implies v_2 \in \text{Couvert}(v_5)$.
- $d(v_6, v_5) = 1 \leq f'(v_6) \implies v_6 \in \text{Couvert}(v_5)$.

Donc : $\text{Couvert}(v_5) = \{v_2, v_6\}$.

Calcul des distances (dm) :

- $dm(v_2) = \max\{d(v_5, v_1), d(v_5, v_3), d(v_5, v_4)\} = \max\{3, 1, 1\} = 3$.
- $dm(v_6) = \max\{d(v_5, v_7)\} = 2$.

Sélection du défenseur (defenseur) :

- pour v_2 : $|f'(v_2) - dm(v_2)| = |2 - 3| = 1$.
- pour v_6 : $|f'(v_6) - dm(v_6)| = |1 - 2| = 1$.

Égalité, choisissons la source ayant le broadcast maximal $f' \implies \text{defenseur} = v_2$.

Mise à jour des broadcast :

- Puisque $f'(v_2) < dm(v_2)$, alors $f''(v_2) = 3$.
- Pour v_6 , le broadcast reste inchangé : $f''(v_6) = f'(v_6) = 1$.

Résultat Final :

$$f(v_2) = 3, \quad f(v_6) = 1 \implies \gamma_{bs}(G) = 4$$

4.2 Analyse de la complexité

Théorème 4.2.1 La complexité temporelle totale de l'algorithme proposé est majorée par :

$$O(n^7 + n^2 + d n).$$

Preuve. L'approche proposée comporte trois phases exécutées de façon séquentielle. La complexité totale de l'algorithme s'obtient donc en sommant les complexités respectives de chacune de ces phases.

Pour la première phase, il s'agit de l'application de l'algorithme de Lokshtanov afin de déterminer un ensemble broadcast dominant. Pour un graphe d'ordre n , cet algorithme s'exécute en temps polynomial, avec une borne supérieure de $O(n^7)$.

Pour la deuxième phase, chaque sommet de broadcast explore son voisinage de taille Δ^f , où Δ représente le degré maximal du graphe et f la valeur de broadcast. Comme chaque sommet calcule les distances entre tous les couples d'adjacents privés (au plus d sommets), la complexité de cette phase est donnée par : $O(d \cdot \Delta^{2f})$.

Comme Δ^{2f} est majoré par n , cette complexité est bornée par : $O(d n)$.

Dans la troisième phase, chaque sommet doit déterminer son défenseur le plus approprié parmi les sources candidates. Cette étape nécessite un temps de calcul de : $O(n^2)$.

Par conséquent, la complexité temporelle totale de l'algorithme est obtenue en additionnant les complexités des trois phases : $O(n^7 + n^2 + d n)$. $\square$

Complexité du problème

La solution retournée n'est qu'une solution approchée. Elle repose sur l'hypothèse qu'un ensemble broadcast dominant reste broadcast dominant sécurisé après modification des broadcasts des sommets. Cependant, ces modifications peuvent entraîner une augmentation importante de la complexité du problème.

Cela nous conduit à proposer la conjecture suivante :

Conjecture 1 *Le problème du secure broadcast domination est un problème NP-complet.*

4.3 Validation et analyse des performances

Dans la suite, nous validons l'approche proposée et évaluons ses performances à l'aide d'instances réelles et synthétiques. Nous présentons d'abord les caractéristiques matérielles et logicielles de l'environnement d'implémentation.

Caractéristiques matérielles et logicielles

Tous les algorithmes de l'approche proposée ont été implémentés en Python, en s'appuyant sur la bibliothèque NetworkX, qui offre une gestion souple et efficace des graphes. Les expérimentations et simulations ont été réalisées sur un ordinateur portable HP EliteBook 840 G3, dont les principales caractéristiques techniques sont les suivantes :

- **Processeur** : Intel(R) Core(TM) i5-6300U CPU @ 2.40 GHz,
- **Mémoire vive (RAM)** : 8 Go,
- **Système d'exploitation** : Windows 10 Professionnel (64 bits).

Instances de test

Instances réelles

Comme mentionné précédemment, la validation de notre approche s'appuie sur des instances réelles et synthétiques. Pour les instances réelles, nous avons utilisé celles de R. Aringhieri et al. [2], qui couvrent différents types de réseaux, comme détaillé dans le tableau ci-dessous :

Instances	$ V $	$ E $
Bovine	121	190
Circuit	252	399
E.coli	328	456
USAir97	332	2126
humanDiseasome	516	1188
Ham1000	1000	1998

TABLE 4.1 – Caractéristiques principales du jeu de données réel

Instances synthétisées

Pour les instances synthétiques, nous générons des graphes de tailles variées selon les modèles les plus couramment utilisés dans la littérature, à savoir :

- **Modèle de Barabási-Albert** Le modèle de Barabási-Albert (BA) est utilisé pour générer des réseaux complexes qui simulent la structure de l'Internet et des réseaux sociaux. Ce processus de génération repose sur deux principes fondamentaux :

- * **La croissance** : Le réseau commence avec un petit nombre de **sommets**, puis de nouveaux **sommets** sont ajoutés un par un de manière continue.
- * **L'attachement préférentiel** : Chaque nouveau **sommet** se connecte à m **sommets** existants avec une probabilité proportionnelle à leur degré (nombre de liens). Mathématiquement, la probabilité Π_i que le nouveau **sommet** se connecte au **sommet** i est donnée par : $\Pi_i = \frac{k_i}{\sum_j k_j}$, où k_i est le degré du **sommet** i .

Dans cette étude, nous avons généré quatre réseaux de tailles différentes ($n \in \{500, 1000\}$) via la fonction `barabasi_albert_graph(n, m)` de la bibliothèque **NetworkX**. En fixant $m = 1$, chaque nouveau **sommet** crée un seul lien, favorisant ainsi une structure de type "Scale-free" où quelques **sommets** (hubs) deviennent très populaires et fortement connectés, reproduisant fidèlement les propriétés topologiques des réseaux réels.

- **Modèle de Forest Fire (Modèle de Propagation)**

Le modèle de Forest Fire (FF) est un modèle génératif conçu pour capturer des propriétés dynamiques observées dans les réseaux réels, telles que la densification et la diminution du diamètre au cours du temps. Le processus de génération repose sur une analogie de "propagation d'incendie" et suit une logique récursive :

- * **Processus de génération** : Lorsqu'un nouveau **sommet** v rejoint le réseau, il choisit d'abord un **sommet** existant w (appelé "ambassadeur") auquel il se connecte. Le "feu" se propage ensuite de w à ses voisins. Le nouveau **sommet** v crée des liens avec une partie des voisins de w , puis avec les voisins de ces voisins, et ainsi de suite de manière récursive.
- * **L'Probabilité (Paramètre de brûlage)** : Le mécanisme est régi par une **probabilité de propagation** p (appelée *forward burning probability*). À chaque étape, le nombre de voisins qu'un **sommet** peut "brûler" (auxquels v va se connecter) est déterminé par une distribution géométrique de paramètre p .

Dans cette étude, nous avons généré quatre réseaux de tailles progressives (250, 500 **sommets**) via la fonction `forest_fire_graph(n, p)` de la bibliothèque **NetworkX**. Ce modèle produit des graphes qui présentent à la fois des propriétés de "petit monde" (*Small-world*) et des distributions de degrés en loi de puissance (*Power-law*), ce qui en fait un environnement de test très réaliste pour notre algorithme de sécurisation.

- **Modèle de Watts-Strogatz (Petit-monde)**

Le modèle de **Watts-Strogatz (WS)** est utilisé pour générer des réseaux de type « petit-monde » (*Small-world*), caractérisés par un fort coefficient de regroupement et des distances courtes entre les **sommets**. Le processus de génération suit une logique de transformation en deux étapes :

- * **Processus de génération** : L'algorithme commence par créer une structure régulière (un anneau) où chaque **sommet** est connecté à ses k voisins les plus proches. Ensuite, il procède à une étape de « re-câblage » (*rewiring*).
- * **La probabilité (Paramètre de re-câblage)** : Pour chaque arête du graphe initial, il existe une **probabilité** p qu'elle soit déconnectée de son voisin actuel pour être reliée aléatoirement à un autre **sommet** choisi uniformément dans le réseau.

Dans cette étude, nous avons généré des réseaux de 250, 500 et 1000 **sommets** via la fonction `watts_strogatz_graph(n, k, p)` de la bibliothèque **NetworkX**. Ce mécanisme permet de passer d'un réseau très ordonné (quand $p = 0$) à un réseau totalement aléatoire (quand $p = 1$), offrant ainsi une structure intermédiaire idéale pour tester la robustesse de notre algorithme sur des réseaux fortement groupés.

Le tableau suivant présente les caractéristiques des différentes instances utilisées, telles qu'elles ont été décrites par Ventresca et al. [31] :

Instances	$ V $	$ E $
BAC500	500	499
BAC1000	1000	999
FF250	250	214
FF500	500	828
WS250	250	1246
WS500	500	1496
WS1000	1000	4996

TABLE 4.2 – Principales caractéristiques du jeu de données synthétique

4.4 Évaluation de la fiabilité et de l'efficacité de l'approche

Dans cette section, nous évaluons la fiabilité de l'approche proposée sur des instances réelles et synthétiques, en comparant les valeurs de γ_b et γ_{bs} . Nous illustrons également l'efficacité de notre méthode à travers l'analyse des temps d'exécution. Le tableau suivant présente les résultats expérimentaux obtenus sur les instances synthétiques.

Instance	γ_b	γ_{bs}	Temps d'exécution		
			Première phase	Deuxième et troisième phases	Temps total
BA n500	7	13	8.71	0.499	9.209
BA n1000	9	18	162.874	1.823	164.697
FF n250	7	14	7.426	0.140	7.566
FF n500	8	15	29.53	0.564	30.064
WS n250	4	6	4.487	0.125	4.612
WS n500	7	10	18.692	0.609	19.301
WS n1000	6	7	117.64	2.696	120.336

TABLE 4.3 – Comparaison de γ_b , γ_{bs} et du temps d'exécution sur des instances synthétiques

Par ailleurs, les résultats obtenus sur les instances réelles, présentés dans le tableau 4.4, confirment la robustesse et l'efficacité de l'approche proposée :

Instance	γ_b	γ_{bs}	Temps d'exécution		
			Première phase	Deuxième et troisième phases	Temps total
Bovine	4	8	2.540	0.062	2.602
Circuit	8	13	6.477	0.124	6.601
Ecoli	7	12	14.140	1.70	15.84
Hamilton1000	7	9	199.380	2.08	201.46
HumanDiseasome	8	15	32.925	0.642	33.567
Usair97	3	6	6.623	0.420	7.043

TABLE 4.4 – Valeurs de γ_b , γ_{bs} et temps d'exécution sur les instances réelles

L'histogramme de la figure 4.2 illustre la comparaison entre le nombre de broadcast

domination γ_b et le nombre de broadcast domination sécurisée γ_{bs} pour les instances synthétiques.

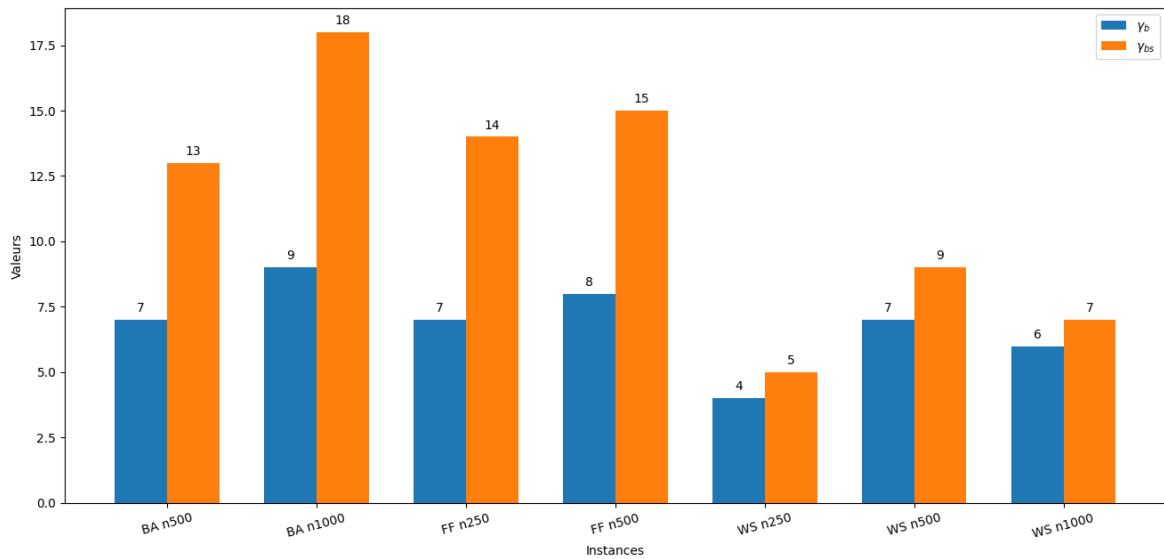


FIGURE 4.2 – Comparaison des paramètres γ_b et γ_{bs} pour les instances synthétiques.

La figure 4.3 montre que le temps d'exécution des deuxième et troisième phases reste nettement inférieur à celui de la première phase, ce qui confirme l'efficacité de la procédure de sécurisation proposée ainsi que son faible coût computationnel par rapport à la phase de construction de l'ensemble de broadcast domination initial.

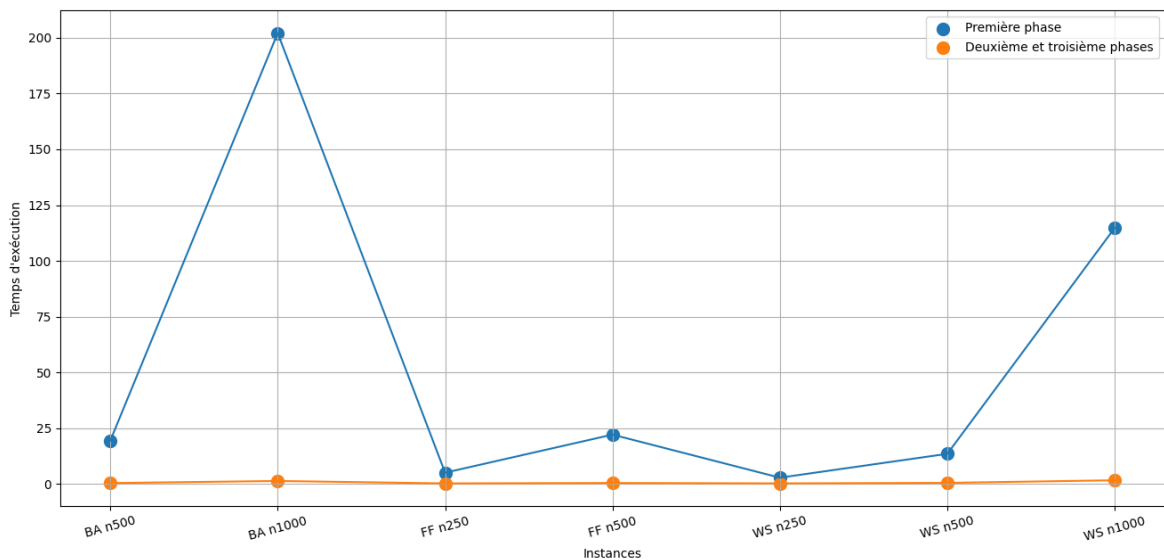


FIGURE 4.3 – Analyse du temps d'exécution par phase (Instances synthétiques)

la figure 4.4 illustre l'efficacité de la phase de calcul de l'ensemble broadcast dominant. Par ailleurs, l'analyse suivante porte sur la comparaison des paramètres γ_b et γ_{bs} pour les instances réelles du jeu de données.

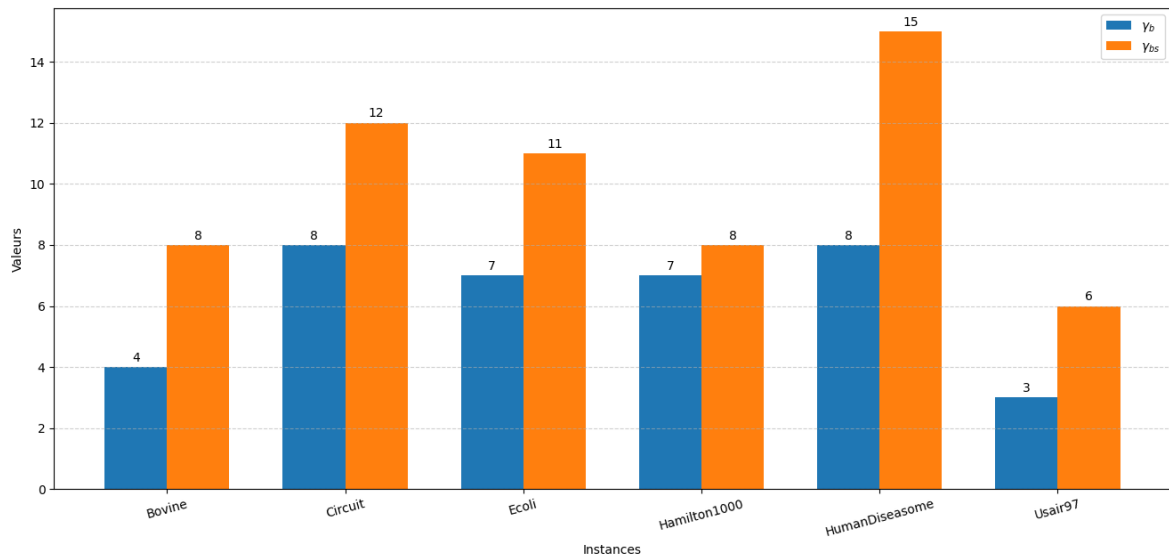


FIGURE 4.4 – Comparaison des paramètres γ_b et γ_{bs} pour les instances réelles.

Les résultats de la figure 4.5 confirment que le temps requis pour les phases de sécurisation (2 et 3) est négligeable par rapport au temps de calcul initial. Cette observation est cruciale, car elle prouve que notre approche peut être appliquée à des réseaux de grande taille sans surcharger les ressources computationnelles.

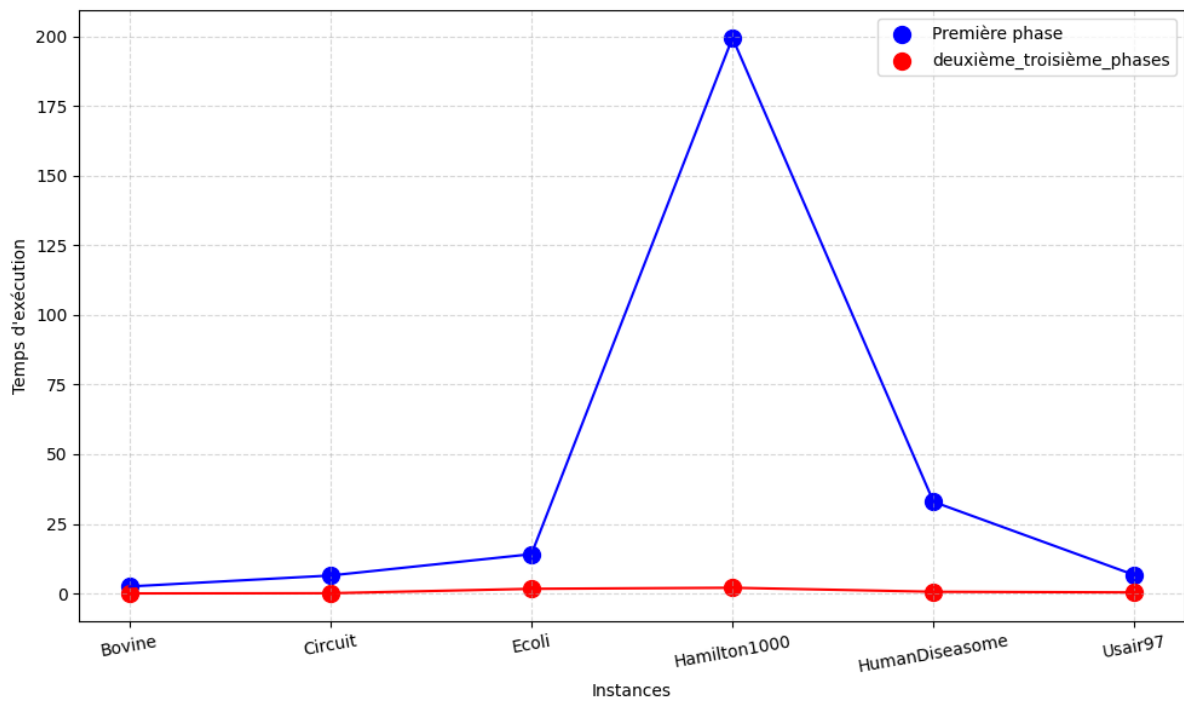


FIGURE 4.5 – Analyse du temps d'exécution par phase (Instances réelles)

Conclusion

Dans ce chapitre, nous avons proposé une approche permettant d'obtenir des solutions quasi-optimales au problème du *Secure Broadcast Dominating Set*. Nous avons également évalué sa fiabilité et son efficacité à l'aide d'instances réelles et synthétiques.

Nous avons émis l'hypothèse que le problème est NP-complet pour les graphes généraux. Cependant, la complexité demeure ouverte pour certaines classes particulières de graphes, telles que les graphes cordaux, les graphes d'intervalle et les graphes d'intervalle propre, que nous prévoyons d'explorer dans de futurs travaux.

Des investigations complémentaires pourraient permettre d'appliquer l'approche proposée à l'étude des topologies orientées sécurité dans les réseaux de capteurs et les réseaux IoT, en considérant les sommets mobiles comme des sommets ayant un broadcast positif.

Bibliographie

- [1] N. Alon and J. H. Spencer, *The Probabilistic Method*, New York, NY, USA : Wiley, 1992.
- [2] R. Aringhieri, A. Grosso, P. Hosteins, and R. Scatamacchia, "A General Evolutionary Framework for Different Classes of Critical Node Problems," *Engineering Applications of Artificial Intelligence*, vol. 55, pp. 128–145, 2016.
- [3] S. Arumugam, K. Ebadi, and M. Manrique, "Co-Secure and Secure Domination in Graphs," National Centre for Advanced Research in Discrete Mathematics.
- [4] C. Berge, *Graphes et Hypergraphes*, Paris, France : Dunod, 1970.
- [5] C. Berge, *Graphs and Hypergraphs*, Amsterdam, Netherlands : North-Holland, 1973.
- [6] M. Blidia, M. Chellali, T. W. Haynes, and M. A. Henning, "Independent and Double Domination in Trees," *Utilitas Mathematica*, vol. 67, pp. 161–171, 2005.
- [7] I. Boufelgha, *Broadcast Domination dans les Prismes Complémentaires*, Mémoire de Magister, USTHB, Alger, Algérie, 2013.
- [8] R. C. Brigham, J. R. Carrington, and R. P. Vitray, "Connected Graphs with Maximum Total Domination Number," *Journal of Combinatorial Theory, Series B*, vol. 38, pp. 101–108, 1985.
- [9] R. C. Brigham, P. Z. Chinn, and R. D. Dutton, "Vertex Domination-Critical Graphs," *Networks*, vol. 18, pp. 173–179, 1988.
- [10] E. W. Chambers, B. Kinnersley, N. Prince, and D. B. West, "Extremal Problems for Roman Domination," 2010.
- [11] M. Chellali and T. Haynes, "A Note on the Total Domination Numbers of Trees," *Discrete Mathematics*, vol. 238, pp. 33–43, 2001.

- [12] M. Chellali and T. Haynes, "On Paired and Double Domination in Graphs," *Utilitas Mathematica*, vol. 67, pp. 161–171, 2005.
- [13] E. J. Cockayne, R. M. Dawes, and S. T. Hedetniemi, "Total Domination in Graphs," *Networks*, vol. 10, pp. 211–219, 1980.
- [14] E. J. Cockayne, B. Gamble, and B. Shepherd, "An Upper Bound for the k -Domination Number of a Graph," *Journal of Graph Theory*, vol. 9, pp. 533–534, 1985.
- [15] E. J. Cockayne, P. J. P. Grobler, W. R. Grundlingh, J. Munganga, and J. H. Van Vuuren, "Protection of a Graph," *Utilitas Mathematica*, vol. 67, pp. 19–32, 2005.
- [16] J. E. Dunbar, D. J. Erwin, T. W. Haynes, S. M. Hedetniemi, and S. T. Hedetniemi, "Broadcasts in Graphs," *Discrete Applied Mathematics*, vol. 154, pp. 59–75, 2006.
- [17] D. J. Erwin, *Cost Domination in Graphs*, Ph.D. Dissertation, Western Michigan University, 2001.
- [18] D. J. Erwin, "Dominating Broadcasts in Graphs," *Bulletin of the ICA*, vol. 42, pp. 89–105, 2004.
- [19] L. Euler, "Solutio Problematis ad Geometriam Situs Pertinentis," *Commentarii Academiae Scientiarum Imperialis Petropolitanae*, vol. 8, pp. 128–140, 1736.
- [20] O. Favaron, "Two Relations Between the Parameters of Independence and Irredundance," *Discrete Mathematics*, vol. 70, pp. 17–20, 1988.
- [21] J. F. Fink and M. S. Jacobson, " n -Domination in Graphs," *Proceedings of the Fifth International Conference*, pp. 283–300, 1984.
- [22] F. Harary and T. W. Haynes, "Double Domination in Graphs," *Ars Combinatoria*, vol. 55, pp. 201–213, 2000.
- [23] T. W. Haynes, S. T. Hedetniemi, and P. J. Slater, *Fundamentals of Domination in Graphs*, New York, NY, USA : Marcel Dekker, 1998.
- [24] L. W. Haynes, M. A. Henning, and P. J. Slater, "Strong Equality of Upper and Independence in Trees," *Utilitas Mathematica*, vol. 59, pp. 111–124, 2001.
- [25] T. W. Haynes, "The Complementary Product of Two Graphs," East Tennessee State University, 2007.
- [26] M. A. Henning, "Graphs with Large Total Domination Number," *Journal of Graph Theory*, vol. 35, pp. 1–11, 2000.

- [27] M. A. Henning and S. T. Hedetniemi, "Defending the Roman Empire – A New Strategy," *Discrete Mathematics*, vol. 266, pp. 239–251, 2003.
- [28] D. Lokshtanov and P. Heggernes, "Optimal Broadcast Domination in Polynomial Time," pp. 4–18, 2005.
- [29] G. Morel, *Stabilité et coloration des graphes sans P_5* , Thèse de Doctorat, Université de Grenoble, France, 2006.
- [30] O. Ore, *Theory of Graphs*, American Mathematical Society Colloquium Publications, vol. 38, 1962.
- [31] M. Ventresca, "Global Search Algorithms Using a Combinatorial Unranking-Based Problem Representation for the Critical Node Detection Problem," *Computers & Operations Research*, vol. 39, pp. 2763–2775, 2012.
- [32] D. B. West, *Introduction to Graph Theory*, Prentice Hall, 2001.
- [33] J. Yang and J. L. Song, "Trees T with Weak Roman Domination Number Being Equal to Domination Number," *Mathematics in Practice and Theory*, vol. 43, pp. 134–140, 2013.