

Placement optimal de moniteurs dans un réseau pour la tomographie booléenne

A. Burlats¹, P. Schaus¹, C. Pelsser¹

¹ UCLouvain, ICTEAM

Résumé

La tomographie booléenne permet de détecter les pannes dans un réseau et de les situer à partir seulement d'un sous-ensemble de noeuds moniteurs. On dit qu'un réseau est k -identifiable si cet ensemble de moniteurs permet de situer jusqu'à k pannes simultanées sans ambiguïté. Les moniteurs impliquant un coût, il faut faire un compromis entre identifiabilité et nombre de moniteurs. Nous présentons et comparons ici un modèle PLNE et un modèle PPC pour trouver le plus petit ensemble de moniteur de manière à garantir la couverture ou la 1-identifiabilité du réseau.

Mots-clés

Programmation par contraintes, tomographie booléenne, k -identifiabilité, supervision de réseaux, programmation linéaire en nombres entier

Abstract

With boolean tomography it is possible to detect and locate failures in a network with only a subset of nodes as monitors. A network is k -identifiable if the set of monitors is able to detect up to k simultaneous failures without ambiguity. Monitors are costly, thus a trade between identifiability and monitors quantity is necessary. We present and compare a MILP model and a CP model to find the smallest set of monitors such that all failures in a network are covered or identifiable.

Keywords

Constraint programming, boolean tomography, networks monitoring, k -identifiability, mixed-integer linear programming

1 Introduction

Pour assurer le bon fonctionnement d'un réseau, il est important d'être en capacité de le superviser. Il existe plusieurs approches pour cela, parmi elles, la tomographie consiste à combiner des mesures bout-en-bout entre des moniteurs à des méthodes d'inférences afin d'estimer l'état du réseau. L'intérêt de cette méthode est qu'elle ne nécessite qu'un sous-ensemble de moniteurs parmi les noeuds du réseau, permettant ainsi de superviser certaines zones où il est compliqué de placer des moniteurs internes.

La tomographie booléenne se concentre sur la détection et localisation d'échecs dans le réseau (pannes, congestions, ...).

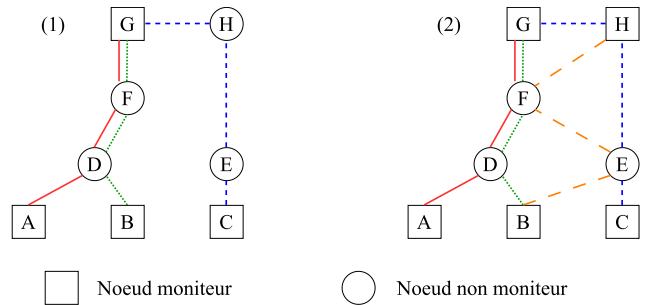


FIGURE 1 – Illustration de la notion de 1-identifiabilité (chaque couleur et style de ligne représente une route) : (A) Tous les noeuds sont couverts mais non 1-identifiables (B) Tous les noeuds sont 1-identifiables.

Dans ce contexte, cette méthode présente un avantage supplémentaire car elle est capable de détecter des pannes plus complexes telles que les *échecs silencieux* [11]. Dans ces situations, bien que tous les équipements soient fonctionnels, la circulation sur le réseau peut être bloquée en raison de conflits entre différents protocoles ou d'erreurs dans leur implémentation.

Concrètement, avec cette approche, les moniteurs s'envoient des messages à travers des *routes de mesures*. Lorsqu'une panne se situe au niveau d'un élément (noeud ou lien), toutes les routes le traversant échouent. La panne va donc être détectée en observant si des routes de mesures ne fonctionnent pas.

Le problème de la couverture consiste à pouvoir détecter la panne de n'importe quel noeud du réseau.

Le problème d'identifiabilité, consiste à pouvoir localiser une panne, et demande donc plus d'informations. Il faut être capable d'identifier la ressource impactée à partir du jeu de routes qui sont tombées et de celles qui restent actives. Suite à la panne d'un noeud, l'ensemble des routes traversant l'élément vont tomber en panne. Ces routes constituent le *symptôme* du noeud. S'il est unique parmi les éléments du réseau, alors on peut conclure que c'est cet élément qui est à l'origine de la panne. On dit d'un réseau qu'il est *1-identifiable* si chaque élément dispose d'un symptôme unique. La figure 1 donne un exemple pour illustrer la 1-identifiabilité. Dans le cas (A) tous les noeuds sont bien couverts car tous sont traversés par une route liant deux moniteurs. Mais si D ou F tombe en panne, dans les deux cas les routes liant A et G ainsi que B et G vont toutes

deux tomber en panne. La panne sera donc détectée, mais il sera impossible de savoir lequel des deux est à l'origine de la panne. Le problème est similaire avec les noeuds E et H. Dans le cas (B) H est un moniteur, offrant une nouvelle route de mesure entre B et H. Ainsi chaque noeud non moniteur dispose d'un symptôme unique : Si D tombe en panne les routes (A,G) et (B,G) vont tomber en panne ; si E tombe en panne ce seront les routes (C,G) et (B,H) ; si F ne fonctionne plus, les routes (A,G), (B,G) et (B,H) ne fonctionneront plus. En notant les routes hors service, il est possible de déterminer sans ambiguïté quel noeud est à l'origine de la panne.

Mais de l'ambiguïté peut apparaître si plusieurs pannes se produisent simultanément car plusieurs symptômes sont alors impliqués. Pour la localisation d'un *ensemble d'échecs*, les nouveaux symptômes sont constitués de l'union des symptômes des noeuds en panne. On dit qu'un réseau est *k-identifiable* si, lorsque que l'on considère tous les ensembles d'au plus *k* éléments possibles, chaque symptôme est unique.

Une forte identifiabilité est logiquement préférable, mais cela peut nécessiter un grand nombre de moniteurs et donc un coût de supervision plus élevé. Dans cet article, nous nous concentrons sur la sélection du plus petit ensemble de moniteurs pour couvrir le réseau ou y garantir la 1-identifiabilité. Une hypothèse importante des réseaux considérés dans cette étude est que les routes entre toutes les paires de nœuds sont considérées comme fixes et imposées par le protocole de routage. Une paire de moniteurs ne peut vérifier que l'état des routes entre eux. En pratique, les opérateurs de réseau configurent généralement les pondérations des liens (IGP) pour influencer le flux de trafic dans le réseau en supposant qu'ils suivent les chemins les plus courts (voir par exemple [4] pour l'optimisation des pondérations IGP). Alternativement, d'autres protocoles tels que le routage de segment ou MPLS [8, 9, 13] permettent d'introduire des déviations ou des configurations de route explicites entre paires de nœuds. Pour tous ces protocoles, les moniteurs sont en mesure de déterminer quels chemins de données entre eux sont affectés par une défaillance.

Dans cet article, nous proposons une formalisation du problème de choix de moniteurs pour assurer la couverture et la 1-identifiabilité. Nous proposons aussi deux modèles pour résoudre ce problème, un modèle de Programmation Linéaire en Nombres Entiers (PLNE) et un modèle de Programmation Par Contraintes (PPC). Enfin nous comparons ces deux modèles entre eux et avec l'état de l'art.

2 Travaux antérieurs

Le problème du placement de moniteur pour la détection de pannes de noeuds a déjà été exploré via d'autres approches que la PPC ou la PLNE. MNMP (Maximum Node-identifiability Monitor Placement) [14] est un algorithme glouton qui ajoute itérativement des moniteurs jusqu'à ce que la *k-identifiabilité* désirée soit atteinte pour ensuite retirer les moniteurs redondants. Malheureusement, il ne garantit pas l'optimalité de la solution. Bezerra et al. [2] pro-

posent différentes améliorations de cet algorithme pour réduire fortement le temps de calcul et le rendre utilisable dans le cas de réseaux sans fils, plus susceptibles de changer régulièrement alors que Bartolini et al. [1] suggèrent une borne supérieure sur le nombre maximal de noeuds identifiables pour un budget de routes de mesures donné. Enfin, Stanic et al. [18] proposent un modèle PLNE et un algorithme glouton pour un problème de choix de moniteurs. Néanmoins le problème considéré y est légèrement différent, ce sont les pannes des liens et non des noeuds qui sont considérées et les routes disponibles sont des allers-retours symétriques, ne nécessitant donc qu'un seul moniteur pour devenir des routes de mesures. Cette approche perd donc en généralité car en réalité les routes dans un réseau ne sont pas forcément symétriques.

3 Formalisation du problème

Dans cet article nous proposons deux modes de résolutions optimaux aux problèmes de couverture et de 1-identifiabilité. Nous présentons ici ces deux problèmes.

3.1 Problème de couverture

Nous supposons que la topologie du réseau est connue, peut être modélisée par un graphe connexe (c.à.d. qu'il existe une route permettant de relier chaque paire de noeuds). Une route ne devient une route de mesure, utilisable pour la tomographie, que si ses noeuds de départ et d'arrivée sont tous les deux des moniteurs. On appelle symptôme d'un noeud l'ensemble des routes de mesure qui le traverse. Le but de ce problème est de choisir parmi les noeuds du réseau lesquels seront moniteurs de manière à :

- s'assurer que chaque noeud soit traversé par au moins une route de mesure.
- minimiser le nombre de moniteurs afin de minimiser les coûts de supervision.

Le problème peut être formalisé ainsi :

Considérons un graphe orienté $G = (V, A)$ où chaque paire de noeuds $(i, j) \in V^2$ dispose d'une route acyclique les reliant. $P(i, j) = \{i, \dots, j\}$ est l'ensemble des noeuds traversés par la route allant de i à j . Les routes ne sont pas forcément symétriques, ainsi $P(i, j)$ et $P(j, i)$ ne contiennent pas nécessairement les mêmes noeuds. Le but est de trouver le plus petit ensemble de noeuds moniteurs M tel que $\bigcup_{(i,j) \in M^2} P(i, j) = V$.

3.2 Problème de 1-identifiabilité

Pour le problème de 1-identifiabilité, nous nous plaçons dans les mêmes conditions que pour la couverture. Ici, notre but est un peu différent car il faut s'assurer que chaque noeud soit identifiable, c'est à dire que si l'un d'entre eux tombe en panne, cette panne doit non seulement être détectable mais aussi localisable sans ambiguïté. Cela implique que chaque noeud soit couvert, ce problème peut donc être vu comme une variante plus contrainte du problème de couverture. Mais pour que la panne soit localisable, il faut que le symptôme lié à la source de cette panne soit unique. Ici, nous nous concentrons sur la 1-identifiabilité des noeuds et ne considérons donc que les symptômes des noeuds.

Le problème peut être formalisé ainsi :

Considérons un graphe orienté $G = (V, A)$ où il existe une route acyclique entre chaque paire de noeuds $(i, j) \in V^2$. $P(i, j) = \{i, \dots, j\}$ est l'ensemble des noeuds traversés par la route allant de i à j . $P(i, j)$ et $P(j, i)$ ne contiennent pas nécessairement les mêmes noeuds étant donné que les routes ne sont pas symétriques. On note S_i le symptôme du noeud i , c'est à dire l'ensemble de routes de mesures qui le traversent. Le but est de trouver le plus petit ensemble de noeuds moniteurs M de sorte à ce que $\bigcup_{(i,j) \in M^2} P(i, j) = V$ (couverture) et que $\forall i \neq j \in V^2, S_i \neq S_j$ (1-identifiabilité).

4 Placement de moniteurs

Ici nous présentons nos deux modèles pour la résolution des problèmes de couverture et de 1-identifiabilité. Nous présentons d'abord un modèle de programmation linéaire en nombres entiers (PLNE), puis un modèle de programmation par contraintes (PPC).

4.1 Modèle PLNE

Notre problème est modélisé à l'aide de deux vecteurs booléens : x , un vecteur de taille $|V|$ modélisant l'ensemble de moniteur (x_i est vrai ssi le noeud i est moniteur) et y , un vecteur de taille $|P|$ modélisant l'ensemble de routes de mesures ($y_{P(i,j)}$ est vrai ssi la route $P(i, j)$ est une route de mesure).

Le but de ce modèle est de limiter le nombre de moniteur, l'objectif est donc :

$$\text{Minimiser } \sum_{i \in V} x_i \quad (1)$$

Une route $P(i, j)$ est une route de mesure ssi son point de départ i et d'arrivée j sont tous deux des moniteurs. Cette relation est modélisée par les 3 contraintes suivantes :

$$y_{P(i,j)} \leq x_i \quad (2)$$

$$y_{P(i,j)} \leq x_j \quad (3)$$

$$y_{P(i,j)} \geq x_i + x_j - 1 \quad (4)$$

Ces 3 contraintes sont appliquées pour chaque route $P(i, j) \in P$. Exiger que tous les noeuds soient couverts signifie que le symptôme de chaque noeud doit être non vide. On note S_i le sous ensemble de P contenant toutes les routes traversant le noeud i , pour qu'un noeud soit couvert il faut donc qu'au moins une route parmi S_i soit une route de mesure. Pour chaque noeud $i \in V$ nous avons donc la contrainte suivante :

$$\sum_{P(i',j') \in S_i} y_{P(i',j')} \geq 1 \quad (5)$$

En s'arrêtant ici, le modèle permet de résoudre le problème de couverture. Mais l'ajout d'un type de contrainte supplémentaire permet de résoudre le problème de 1-identifiabilité. Un noeud est 1-identifiable ssi il est distinguable de chaque autre noeud. Contraindre chaque noeud

à être 1-identifiable revient à contraindre chaque paire de noeud à être distinguables. Pour que deux noeuds soient distinguables il suffit qu'au moins une route de mesure traverse l'un des noeuds sans traverser l'autre. On note $D_{i,j}$ l'ensemble des routes traversant le noeud i sans passer par j et inversement. Pour chaque $i, j \in V^2$ tels que $i < j$, les deux noeuds sont assurés d'être distinguables par la contrainte :

$$\sum_{P(i',j') \in D_{i,j}} y_{P(i',j')} \geq 1 \quad (6)$$

Pour cette contrainte nous avons défini i et j comme étant des noeuds, mais il est possible de considérer des ensembles de noeuds et de résoudre donc le problème de k -identifiabilité pour des valeurs de k plus grandes. Mais le nombre de contraintes augmente alors de manière exponentielle avec k : il faut considérer toutes les combinaisons de noeuds de taille inférieure ou égale à k . On a $n = \sum_{i=1}^k \binom{|V|}{i}$ ensembles à considérer. Et il y a une contrainte (6) par paire d'ensembles possible, soit $\binom{n}{2}$ contraintes. Le nombre de contraintes explose donc rapidement avec k .

4.2 Modèle PPC

Le modèle PPC suit la même logique que le modèle PLNE. Un vecteur de variables booléennes x de taille $|V|$ indique quels noeuds sont moniteurs, un autre vecteur y de taille $|P|$ indique quelles routes sont des routes de mesures. L'objectif est le même que celui exprimé par l'équation (1), c'est-à-dire minimiser la somme des x .

Une route n'est une route de mesure que si ses noeuds de départ et d'arrivée sont des moniteurs. Cette condition est modélisée par une réification de la contrainte ET entre $y_{P(i,j)}$, x_i et x_j , contraignant $y_{P(i,j)} \equiv x_i \wedge x_j$. Cette contrainte force la même relation que les contraintes (2, 3, 4) du modèle PLNE.

Notons S_i l'ensemble des routes qui traversent le noeud i . La contrainte OU force une liste de variables booléennes à contenir au moins une fois la valeur 1. Ainsi pour chaque noeud i , la contrainte $\bigvee_{P(i',j') \in S_i} y_{P(i',j')}$ permet de s'assurer que le noeud soit traversé par au moins une route de mesure. Ceci correspond à la contrainte (5) du modèle PLNE.

Enfin pour la 1-identifiabilité, si l'on note $D_{i,j}$ l'ensemble des routes permettant de distinguer les noeuds i et j , c'est les routes qui passent par l'un des deux noeuds seulement, la contrainte $\bigvee_{P(i',j') \in D_{i,j}} y_{P(i',j')}$ permet de s'assurer qu'au moins une route de mesure permette de différencier i et j . L'appliquer pour chaque paire de noeud $i, j \in V^2$ permet donc d'assurer la 1-identifiabilité. Cette contrainte correspond à l'inégalité (6) du modèle PLNE.

Un noeud traversé par de nombreuses routes est plus susceptible d'être traversé par des routes de mesures et donc moins susceptible d'être un moniteur dans la solution optimale. Ce concept est similaire à la métrique de la centralité d'un noeud, courante dans la théorie des graphes [16]. Nous proposons donc une stratégie de branchement basée sur cette métrique. L'heuristique *centralité-max*, sélectionne le noeud non fixé i qui présente la plus forte cen-

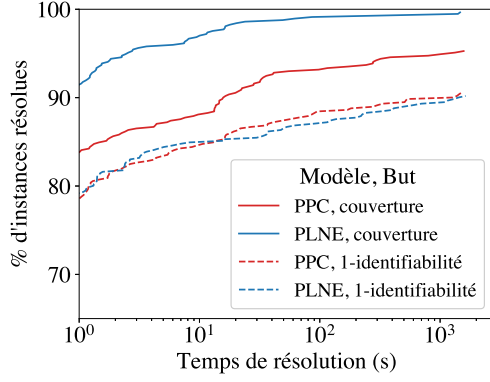


FIGURE 2 – Proportion d’instances résolues en fonction du temps de calcul de la résolution

Modèle	But	Meilleure solution	Optimalité
PLNE	Couv.	571 (100.0%)	571 (100.0%)
	1-id.	570 (99.82%)	569 (99.65%)
PPC	Couv.	565 (98.95%)	544 (95.27%)
	1-id.	563 (98.60%)	525 (91.94%)
MNMP	Couv.	546 (95.62%)	None
	1-id.	454 (79.51%)	None

TABLE 1 – Nombre d’instances où la meilleure solution est obtenue et où l’optimalité de la solution est prouvée pour chaque modèle

tralité et le marque comme non moniteur dans la branche la plus à gauche $x_i = 0$. Cette stratégie peut être combinée avec une recherche basée sur le conflit [5, 12].

5 Résultats expérimentaux

Dans cette section nous cherchons à évaluer nos deux modèles. Nous utilisons Gurobi [7] pour résoudre le modèle PLNE et le solveur MiniCP [15] pour le modèle PPC. Nous exécutons nos modèles sur des topologies venant de *Rocketfuel* [17], du *Internet Topology Zoo* [10], du *Internet Topology Data Kit* de CAIDA [3] datant de février 2022 et de *Repetita* [6]. Pour les topologies non connexes nous ne gardons que la plus grosse composante. Nous avons au total 571 topologies avec un nombre de noeuds allant de 4 à 631. Dans notre étude, nous supposons que la route entre deux noeuds correspond au plus court chemin. En cas d’égalité, une seule route est choisie arbitrairement parmi les plus courts chemins.

La figure 2 compare les temps de calculs nécessaires à la résolution de nos modèles selon le but recherché (assurer la couverture ou la 1-identifiabilité). Le temps de calcul limite a été fixé à 30 minutes. Les courbes montrent la proportion d’instances résolues dans un temps inférieur ou égal à la durée en abscisse. Les temps de calculs inférieurs à 1 seconde ont été tronqués. On observe sur cette figure que pour le problème de couverture le modèle PLNE est plus rapide, mais que pour le problème de 1-identifiabilité les deux modèles donnent des performances similaires.

Afin de constater la plus-value de nos modèles par rapport

à l’état de l’art, nous comparons nos modèles avec notre implémentation de la méthode MNMP [14].

La table 1 recense pour chacun des modèles le nombre d’instances où leur solution correspond à la meilleure trouvée, ainsi que le nombre d’instances pour lesquelles ils prouvent l’optimalité de leur solution. MNMP étant un algorithme glouton, il n’est pas capable de prouver que ses solutions sont optimales mais présente un temps de calcul plus faible (au plus 2 secondes pour le problème de couverture et 365 secondes pour la 1-identifiabilité).

L’algorithme retourne une solution optimale dans 95,62% des cas pour le problème de couverture, et dans 79,51% des cas pour l’objectif de 1-identifiabilité. Bien qu’il soit fiable pour le problème de couverture, nous constatons que, pour plus de 100 topologies, les modèles PPC et PLNE peuvent fournir de meilleures solutions pour l’objectif de 1-identifiabilité. L’amélioration apportée par nos modèles est généralement de l’ordre de 1 à 3 moniteurs. Toutefois, l’ajout d’un moniteur entraîne la nécessité de sonder régulièrement $2 * |M| - 1$ nouvelles routes de mesure, M étant l’ensemble des moniteurs. Par conséquent, lorsqu’un grand nombre en est requis, ajouter quelques moniteurs supplémentaires peut avoir un impact significatif sur le trafic et entraîner une congestion. Il est donc crucial d’économiser leur nombre autant que possible. L’usage des modèles PPC et PLNE est donc pertinent pour ce problème malgré le temps de calcul plus élevé.

6 Conclusion

Nous avons présentés un modèle PLNE et un modèle PPC pour résoudre le problème du placement de moniteur dans un réseau. Leur évaluation sur 571 graphes tirés de topologies réelles montre un intérêt de ces approches de par leur capacité à trouver la solution optimale dans des temps de calculs raisonnables pour la plupart des topologies. Pour la suite, nos deux modèles peuvent être améliorés afin d’en réduire le temps de calcul.

Par exemple, notre modèle PPC n’exploite pas tout le potentiel de la PPC car il n’utilise pas de contraintes globales. Trouver de nouvelles contraintes globales pour ce problème permettrait une réduction plus forte de l’arbre de recherche et améliorerait donc le temps de calcul. Nos modèles pourraient alors être applicables à de plus grandes topologies.

Références

- [1] Novella Bartolini, Ting He, Viviana Arrigoni, Annalisa Massini, Federico Trombetti, and Hana Khamfroush. On Fundamental Bounds on Failure Identifiability by Boolean Network Tomography. *IEEE/ACM Transactions on Networking*, 28(2) :588–601, April 2020. Conference Name : IEEE/ACM Transactions on Networking.
- [2] Pamela Bezerra, Po-Yu Chen, Julie A. McCann, and Weiren Yu. Adaptive Monitor Placement for Near Real-time Node Failure Localisation in Wireless Sensor Networks. *ACM Transactions on Sensor Networks*, 18(1) :2 :1–2 :41, October 2021.

- [3] CAIDA. The caida macroscopic internet topology data kit, February 2022.
- [4] Bernard Fortz and Mikkel Thorup. Internet traffic engineering by optimizing ospf weights. In *Proceedings IEEE INFOCOM 2000. conference on computer communications. Nineteenth annual joint conference of the IEEE computer and communications societies (Cat. No. 00CH37064)*, volume 2, pages 519–528. IEEE, 2000.
- [5] Steven Gay, Renaud Hartert, Christophe Lecoutre, and Pierre Schaus. Conflict ordering search for scheduling problems. In *Principles and Practice of Constraint Programming : 21st International Conference, CP 2015, Cork, Ireland, August 31–September 4, 2015, Proceedings 21*, pages 140–148. Springer, 2015.
- [6] Steven Gay, Pierre Schaus, and Stefano Vissicchio. Repetita : Repeatable experiments for performance evaluation of traffic-engineering algorithms, 2017.
- [7] Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2023.
- [8] Renaud Hartert, Pierre Schaus, Stefano Vissicchio, and Olivier Bonaventure. Solving segment routing problems with hybrid constraint programming techniques. In *Principles and Practice of Constraint Programming : 21st International Conference, CP 2015, Cork, Ireland, August 31–September 4, 2015, Proceedings 21*, pages 592–608. Springer, 2015.
- [9] Renaud Hartert, Stefano Vissicchio, Pierre Schaus, Olivier Bonaventure, Clarence Filsfils, Thomas Telkamp, and Pierre Francois. A declarative and expressive approach to control forwarding paths in carrier-grade networks. *ACM SIGCOMM computer communication review*, 45(4) :15–28, 2015.
- [10] S. Knight, H.X. Nguyen, N. Falkner, R. Bowden, and M. Roughan. The internet topology zoo. *Selected Areas in Communications, IEEE Journal on*, 29(9) :1765–1775, october 2011.
- [11] R. R. Kompella, J. Yates, A. Greenberg, and A. C. Snoeren. Detection and Localization of Network Black Holes. In *IEEE INFOCOM 2007 - 26th IEEE International Conference on Computer Communications*, pages 2180–2188, May 2007. ISSN : 0743-166X.
- [12] Christophe Lecoutre, Lakhdar Saïs, Sébastien Tabary, and Vincent Vidal. Reasoning from last conflict (s) in constraint programming. *Artificial Intelligence*, 173(18) :1592–1614, 2009.
- [13] Youngseok Lee, Yongho Seok, Yanghee Choi, and Changhoon Kim. A constrained multipath traffic engineering scheme for mpls networks. In *2002 IEEE International Conference on Communications. Conference Proceedings. ICC 2002 (Cat. No. 02CH37333)*, volume 4, pages 2431–2436. IEEE, 2002.
- [14] Liang Ma, Ting He, Ananthram Swami, Don Towsley, and Kin K. Leung. On optimal monitor placement for localizing node failures via network tomography. *Performance Evaluation*, 91 :16–37, September 2015.
- [15] L. Michel, P. Schaus, and P. Van Hentenryck. Minicp : a lightweight solver for constraint programming. *Mathematical Programming Computation*, 13(1) :133–184, 2021.
- [16] Mark Newman. *Networks*. Oxford university press, 2018.
- [17] Neil Spring, Ratul Mahajan, and David Wetherall. Measuring ISP topologies with rocketfuel. *ACM SIGCOMM Computer Communication Review*, 32(4) :133–145, August 2002.
- [18] Sava Stanic, Suresh Subramaniam, Gokhan Sahin, Hongsik Choi, and Hyeong-Ah Choi. Active monitoring and alarm management for fault localization in transparent all-optical networks. *IEEE Transactions on Network and Service Management*, 7(2) :118–131, June 2010. Conference Name : IEEE Transactions on Network and Service Management.