

Analyse des performances d’algorithmes DCOP pour l’association d’utilisateurs de réseaux HetNets

G. Picard^{1,2} P. Rust³

¹ Institut Henri Fayol, MINES Saint-Etienne, France

² Univ Lyon, CNRS, Laboratoire Hubert Curien UMR 5516, Saint Etienne, France

³ Orange Labs, France

picard@emse.fr
pierre.rust@orange.com

Résumé

Cet article étudie l’applicabilité des techniques d’optimisation sous contraintes distribuée (DCOP) dans le cadre des réseaux radios hétérogènes (HetNets). Nous nous intéressons plus particulièrement au problème de l’association des terminaux utilisateurs aux stations de base, dont l’objectif est de déterminer quel utilisateur (e.g. téléphone mobile) est associé à quelle station de base pour créer un canal de communication. Toutes les stations ne garantissent pas les mêmes qualités de service et d’expérience aux utilisateurs, et la qualité de l’allocation finale (e.g. le débit total obtenu) est un critère à maximiser. Nous présentons ici un modèle DCOP de ce problème d’association, qui se base sur une agentification des stations de base se coordonnant pour obtenir une allocation efficace. Nous évaluons les performances d’algorithmes légers (MGM, MGM-2, DSA) sur un environnement simulé réaliste. Leurs performances sont comparées à celles d’une solution optimale, ainsi qu’à celle de la politique d’association actuellement utilisée dans ces réseaux, à savoir maxSINR.

Mots-clés

DCOP, HetNet, Allocation de ressources, SINR

Abstract

This paper studies the applicability of distributed constrained optimization techniques (DCOP) in the context of heterogeneous radio networks (HetNets). In particular, we are interested in the problem of associating user terminals to base stations, the objective of which is to determine which user (e.g. mobile phone) is associated to which base station to create a communication channel. Not all stations guarantee the same quality of service and experience to users, and the quality of the final allocation (e.g. the total throughput obtained) is a criterion to be maximized. We present here a DCOP model of this association problem, which is based on an agentification of base stations coordinating to obtain an efficient allocation. We evaluate the performance of lightweight algorithms (MGM, MGM-2, DSA) on a realistic simulated environment. Their performances are compared to those of an optimal solution,

as well as to those of the association policy currently used in these networks, i.e. maxSINR.

Keywords

DCOP, HetNet, Resource allocation, SINR

1 Introduction

Dans le contexte des réseaux de communication 5G, l’infrastructure sans fil est constituée de multiples stations de base (antennes émettrices et réceptrices) auxquels les utilisateurs du réseau sont associés pour obtenir un canal de communication et ainsi se connecter au réseau et à Internet. Les stations de base ayant des propriétés différentes (puissance, gain, portée, etc.), on parle alors de réseaux hétérogènes, ou *HetNet*. La plupart des infrastructures cibles considèrent des réseaux à plusieurs tiers –e.g. 3 tiers, avec des stations *macro* à longue portée, des stations *pico* à moyenne portée, et des stations *femto* à courte portée, comme illustré dans la Figure 1. Associer les utilisateurs à des stations de base est un problème d’allocation difficile, dont les objectifs peuvent être de maximiser le signal reçu par chaque utilisateur (pour maximiser sa qualité d’expérience), et de répartir la charge sur les stations de base afin de ne pas réduire le débit par station. Compte tenu de la mobilité des usagers, et donc de la nécessité de fournir des solutions de manière réactive, les approches par optimisation centralisée sont presque inapplicables dans ce contexte.

Nous proposons ici d’évaluer les performances des algorithmes d’optimisation sous contraintes distribuée (DCOP) à ce problème, naturellement distribuable et modélisable comme un système multi-agent. Ici, les stations de base se coordonnent via un algorithme d’optimisation sous contraintes distribuée, pour déterminer quels utilisateurs elles vont servir. L’objectif ici est de maximiser globalement le débit total fourni aux usagers. L’approche classique pour résoudre de tels problèmes d’allocation, pour évaluer la qualité des autres approches, est de suivre une approche par programmation mathématique et optimisation en nombres entiers [11]. Du point de vue applicatif, compte tenu de la difficulté à résoudre de tels problèmes en temps

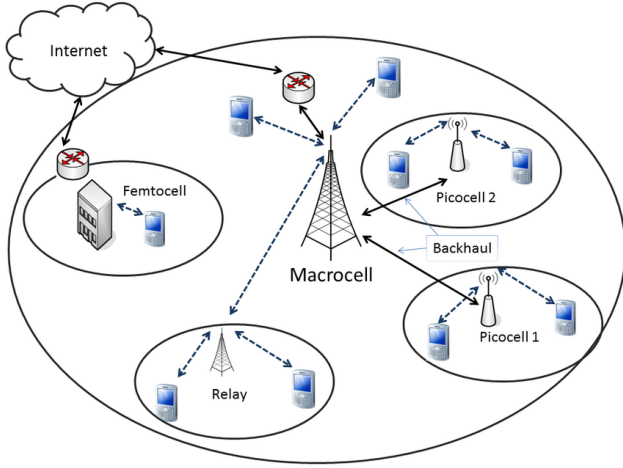


FIGURE 1 – Architecture d’un réseau radio hétérogène à trois tiers (macro, pico, femto) [4]

réel, des approches heuristiques sont utilisées, et principalement l’approche gloutonne maxSINR [12] qui consiste à associer le terminal à la station exhibant le meilleur rapport signal sur interférence plus bruit (SINR). Cependant, cette approche a le désavantage d’associer les usagers en priorité aux stations de hauts tiers (e.g. macro), et ainsi les surcharger, provoquant éventuellement des problèmes de connexion. A notre connaissance, seul [8] propose une approche par DCOP pour s’attaquer à ce problème d’allocation, mais fait le choix d’un algorithme complet, BnB-ADOPT, sans évaluer les performances d’algorithmes plus légers, comme MGM ou DSA. Nous proposons ainsi, dans cette étude, d’élargir nos connaissances sur l’applicabilité des DCOPs à ce problème.

Cet article est structuré comme suit. Le modèle et les algorithmes DCOP principalement utilisés dans la suite sont brièvement introduits en Section 2. Le problème d’association d’utilisateurs dans les HetNets, ainsi que le modèle du système sont présentés en Section 3. La modélisation de ce problème dans le formalisme DCOP y est également détaillé. La section 4 expose le protocole expérimental ainsi qu’une analyse des résultats obtenus sur un environnement simulé réaliste suivant les scénarios proposés dans la littérature du domaine [8, 2]. Enfin, l’article se conclut autour des perspectives offertes par cette investigation en Section 5.

2 Optimisation sous contraintes distribuée

Cette section introduit brièvement le cadre des DCOPs, ainsi que les algorithmes que nous allons utiliser.

2.1 Modèle canonique

Une façon de modéliser un problème distribué d’allocation de ressources est de le formaliser comme un problème d’optimisation distribuée sous contraintes (DCOP) [18].

Définition 1. Un problème d’optimisation sous contraintes

distribué (ou DCOP pour Distributed Constraint Optimization Problem) est un tuple $P = \langle \mathcal{A}, \mathcal{X}, \mathcal{D}, \mathcal{C}, \mu, f \rangle$, où : $\mathcal{A} = \{a_1, \dots, a_m\}$ est l’ensemble d’agents ; $\mathcal{X} = \{x_1, \dots, x_n\}$ sont les variables appartenant aux agents ; $\mathcal{D} = \{\mathcal{D}_{x_1}, \dots, \mathcal{D}_{x_n}\}$ est un ensemble de domaines finis tels que la variable x_i prend ses valeurs dans $\mathcal{D}_{x_i} = \{v_1, \dots, v_{|\mathcal{D}_{x_i}|}\}$; $\mathcal{C} = \{c_1, \dots, c_k\}$ est un ensemble de contraintes souples, où chaque c_i définit une utilité $\in \mathbb{R} \cup \{-\infty\}$ pour chaque combinaison d’affectation de valeurs au sous-ensemble de variables impliquées ; $\mu : \mathcal{X} \rightarrow \mathcal{A}$ est une application associant les variables à leur agent. $f : \prod \mathcal{D}_{x_i} \rightarrow \mathbb{R}$ est une fonction objectif, représentant le coût global d’une affectation de valeurs aux variables.

L’objectif d’optimisation est représenté par la fonction f , qui peut être de différente nature. En général, cette fonction est une somme des coûts des contraintes : $f = \sum_i c_i$.

Définition 2. Une solution à un DCOP P est une affectation de valeurs à toutes les variables. Une solution est dite optimale si elle maximise f .

Notons que la notion d’utilité peut être substituée par la notion de coût $\in \mathbb{R} \cup \{+\infty\}$. Dans ce cas, résoudre un DCOP consiste à minimiser la somme totale des coûts. De plus, il est possible d’intégrer des contraintes dures (comme dans le cas de la satisfaction) en leur attribuant un coût infini en cas de violation.

Comme souligné par [10, 19], les DCOP ont été largement étudiés et appliqués à de nombreux domaines de référence. Ils présentent de nombreuses propriétés intéressantes : (i) focus sur des approches décentralisées où les agents négocient une solution jointe au travers d’échanges locaux de messages ; (ii) exploitation de la structure du domaine (en l’encodant dans des contraintes) pour s’attaquer à des problèmes computationnels difficiles ; (iii) grande variété de méthodes de résolution allant de méthodes exactes à des techniques heuristiques et approchées ; comme, par exemple, ADOPT [17], DPOP [18], MaxSum [9], DSA [25] ou MGM [15], pour ne citer que les plus connus.

2.2 Algorithmes

De nombreux algorithmes de résolution de DCOP existent et se distinguent par leur type de processus de résolution (recherche/inférence), leur complétude (optimal/approché), ou leur parallélisme (synchrone/asynchrone) [10]. De plus, certains algorithmes vont être capables de gérer mieux que d’autres certains types de contrainte ; e.g. les algorithmes par inférence, comme MaxSum et DPOP, nécessitent une mémoire exponentielle dans l’arité des contraintes, donc seront peu adaptés à des problèmes présentant des contraintes qui impliquent un grand nombre de variables. Nous allons utiliser dans cet article deux algorithmes approchés classiques, et très légers (en charge de calcul et en quantité d’information échangée), à savoir DSA [25] et MGM [15].

DSA. Plus qu’un simple algorithme, DSA (*Distributed Stochastic Algorithm*) est une famille d’algorithmes incomplets de recherche locale, très légers, basée sur une idée

assez simple : les agents partent d'une valeur aléatoire de leur domaine et évaluent régulièrement si la qualité de leur propre affectation partielle, définie comme la somme des contraintes dans lesquelles elle est impliquée, pourrait être améliorée en sélectionnant une nouvelle valeur [25]. Cette évaluation est basée sur la connaissance des valeurs actuellement sélectionnées par leurs voisins. Si cette qualité peut être améliorée, l'agent décide aléatoirement, avec une probabilité d'activation p , de sélectionner la valeur correspondante et d'envoyer son état actualisé à ses voisins. Ce processus de recherche est local, car les agents ne fondent leur décision que sur leur connaissance des valeurs de leurs voisins directs. Bien sûr, cet algorithme de recherche locale peut être piégé dans un minimum local (même si la stochasticité de DSA l'aide parfois à échapper à ce minimum local) et ne garantit pas de trouver la solution optimale. Bien que ce ne soit pas strictement le cas, la DSA est un algorithme itératif et peut être utilisée pour obtenir une affectation complète à tout moment, en temps réel, avec une qualité de solution s'améliorant, en moyenne, au fil du temps. Cependant, dans le cas général, DSA ne fournit aucune garantie de monotonie : comme il n'y a pas de coordination dans le processus de décision et que les connaissances locales d'un agent peuvent être dépassées, deux agents peuvent prendre simultanément des décisions contradictoires, ce qui entraîne une diminution de la qualité du résultat global.

Cinq variantes – à savoir DSA-A, DSA-B, DSA-C, DSA-D et DSA-E – de ce principe de base ont été étudiées [25], en fonction de la stratégie utilisée pour le changement de valeurs. Un agent peut sélectionner une nouvelle valeur de manière plus ou moins agressive, lorsque la qualité de son état peut être améliorée, strictement ou non, et lorsque des conflits subsistent même si la qualité ne peut être améliorée. Ces variantes présentent divers degrés de parallélisme et d'exploration de l'espace des solutions. L'approche DSA-B est considérée comme la plus efficace dans le cas général, mais nécessite une pré-exploration de l'espace des solutions. Il a également été démontré dans [25] que la valeur utilisée pour la probabilité d'activation p a une influence considérable sur l'efficacité et la qualité de DSA et présente une propriété de transition de phase. Lorsque la bonne variante et la bonne probabilité d'activation ont été sélectionnées pour une classe de problèmes donnée, DSA fournit des résultats de très bonne qualité, avec une charge de réseau et de calcul minimale, ce qui la rend très évolutive. Il convient de noter que la DSA est capable de travailler avec des contraintes n -aires sans aucune modification.

MGM. MGM (*Maximum Gain Message*) est une modification de l'algorithme DBA qui se concentre sur la transmission du message à gain maximal [15, 14]. Comme le DSA, MGM est un algorithme de recherche locale incomplète qui peut gérer des contraintes n -aires. MGM est un algorithme synchrone : à chaque tour, les agents calculent le changement maximal de qualité, appelé *gain*, qu'ils pourraient obtenir en sélectionnant une nouvelle valeur et envoient ce gain à leurs voisins. Un agent n'est alors autorisé à chan-

ger sa valeur que si son gain est supérieur à celui reçu de tous ses voisins. Ce mécanisme garantit que deux variables impliquées dans la même contrainte ne changeront jamais leur valeur dans le même cycle. Ce processus se répète jusqu'à ce qu'une condition de terminaison soit remplie. Bien qu'il ne fournisse aucune limite à la qualité de la solution, MGM est en mesure de garantir la monotonie ; l'élimination de l'aspect stochastique de la DSA garantit que la qualité de la solution ne s'améliore qu'avec le temps. La monotonie est une qualité très intéressante dans de nombreux domaines d'application, cependant, cette qualité est garantie au détriment d'une tendance plus élevée à se retrouver piégée dans un minima local. Pour atténuer ce problème, [15] propose une version coordonnée de MGM (généralement MGM-2, mais elle peut être étendue à MGM- k), où k agents peuvent coordonner un changement simultané (c'est-à-dire dans le même cycle) de valeurs. Cela permet d'éviter certains minima locaux tout en préservant la monotonie de l'algorithme.

3 Modèle du problème d'association

Cette section introduit le problème d'association dans les HetNet et détaille le modèle du système ainsi que sa physique. Une formulation DCOP est alors proposée pour ce problème.

3.1 Problème d'association d'utilisateurs

Le problème d'association d'utilisateurs peut être formulé comme un problème d'allocation de ressources distribuées dans lequel une ressource unitaire se réfère à un bloc de ressources (RB) qui couvre une certaine plage de fréquence et une certaine durée [26]. Pour obtenir une bonne qualité de service, un utilisateur devra réserver plusieurs blocs sur une station, dans la limite disponible. En effet, la qualité de service est proportionnelle au nombre de blocs réservés. Le nombre de blocs à réserver dépend de la puissance de la station et de la position de l'utilisateur par rapport à la station, à cause des pertes de signal.

Ces dernières années, des efforts de recherche importants ont été consacrés au développement de méthodes distribuées, en particulier des méthodes fondées sur la théorie des jeux et des méthodes d'optimisation combinatoire [3, 13, 11, 24]. Ici, la formulation du modèle de système s'inscrit dans le cadre de la programmation mixte non convexe en nombres entiers [11], qui est toujours NP-difficile. La manière d'obtenir la solution est souvent très coûteuse. Les résultats de simulation dans [3] et [24] montrent que la performance des méthodes dans les schémas basés sur la théorie des jeux et l'optimisation combinatoire est meilleure que certaines méthodes géométriques, par exemple maxSINR. Cependant, ces deux approches – théorie des jeux, et optimisation combinatoire – sont limitées par le fait qu'un HetNet est un système dynamique dans lequel le nombre d'utilisateurs et la configuration de la ressource varient fréquemment dans les stations de base.

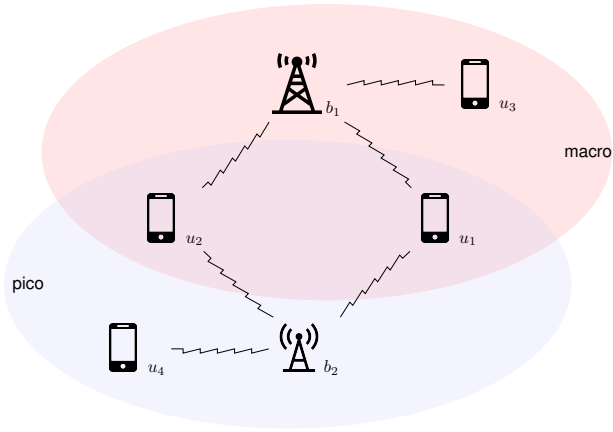


FIGURE 2 – Exemple de configuration de HetNet, avec 4 utilisateurs et 2 stations de base sur 2 tiers

3.2 Modèle du système

Reprenant le modèle de [8], un tier dans le HetNet indique une macrocellule, une picocellule ou une femtocellule où chaque niveau contient un ensemble de stations de base ayant les mêmes configurations (puissance d'émission et ressource). La ressource configurée au niveau d'une station de base fait référence à un ensemble de blocs de ressources (RB) où chaque RB est constitué d'une certaine durée et d'une certaine largeur de bande [7]. Considérons un HetNet à k tiers comprenant des stations de base $\mathcal{B} = \{b_1, \dots, b_{|\mathcal{B}|}\}$ et des utilisateurs $\mathcal{U} = \{u_1, \dots, u_{|\mathcal{U}|}\}$, comme illustré en Figure 2. Notons N_i le nombre de blocs maximal sur b_i . En supposant que les informations sur l'état du canal soient disponibles au niveau des stations de base, les stations de base de différents niveaux se partagent également la largeur de bande totale, de sorte qu'il y a des brouillages à la fois intra- et inter-niveaux lorsque les stations de base attribuent instantanément des blocs aux utilisateurs. Le rapport signal sur bruit plus interférence (SINR) vécu par l'utilisateur u_j servi par b_i dans le k ième tier, illustré en Figure 3, est donné par :

$$\text{SINR}_{ij} = \frac{P_k g_{ij}}{\sum_{b_l \neq b_i} P_k g_{lj} + BN_0} \quad (1)$$

où P_k est la puissance d'émission pour les stations de base du k ième tier, g_{ij} est le gain de puissance du canal entre u_j et b_i , B est la largeur de bande et N_0 est la densité spectrale de puissance de bruit. Le gain de puissance du canal comprend l'effet de l'affaiblissement du signal et son évanouissement. On suppose que l'affaiblissement sur le trajet est statique et que son effet est pris en compte dans la valeur moyenne du gain de puissance du canal, tandis que les évanouissements sont supposés suivre la distribution exponentielle. Ensuite, l'efficacité de l'utilisateur u_j associé à la station b_i , notée e_{ij} est :

$$e_{ij} = \log_2(1 + \text{SINR}_{ij}) \quad (2)$$

En considérant la largeur de bande B , la durée T et l'intervalle de planification Γ configurés pour chaque RB, on

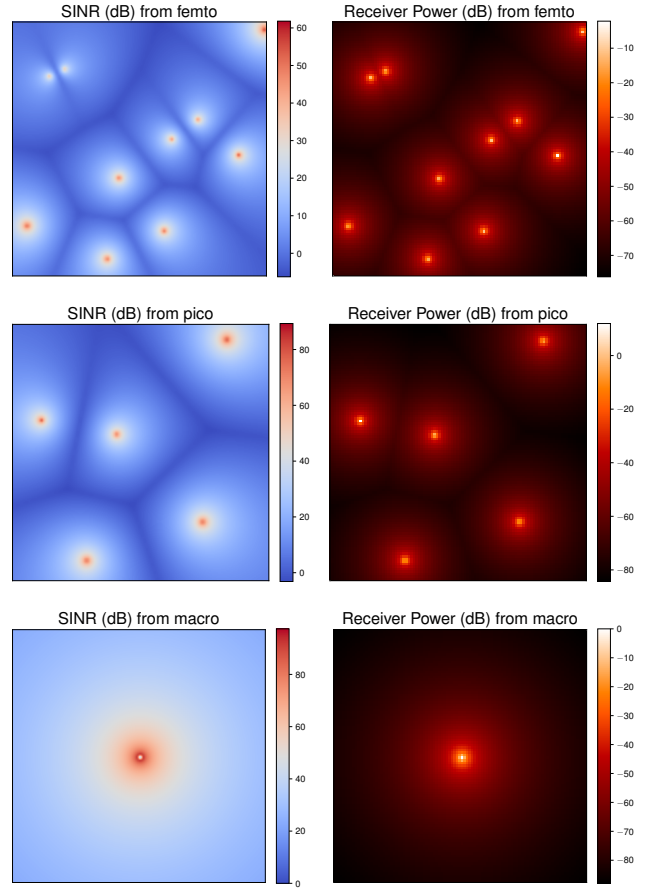


FIGURE 3 – SINR et puissance reçus en chaque point de l'environnement pour les différents tiers d'une infrastructure

obtient le taux unitaire à u_j sur un RB suivant :

$$q_{ij} = \frac{BTe_{ij}}{\Gamma} \quad (3)$$

et ainsi le taux reçu par u_j avec n_{ij} blocs réservés sur b_j du k ième tier est :

$$r_{ij} = n_{ij}q_{ij} \quad (4)$$

La qualité de service (QoS) de chaque utilisateur est alors exprimée comme le taux total minimum que l'utilisateur pourra obtenir. En posant γ le taux minimal accepté par les utilisateurs (e.g. on n'accepte que des services à 50 Mbps), nombre minimum de blocs requis est estimé à :

$$n_{\min}^{ij} = \lceil \frac{\gamma}{q_{ij}} \rceil \quad (5)$$

Maintenant que le modèle physique et les définitions du HetNet ont été posées, nous présentons une formulation DCOP de ce problème.

3.3 Formulation DCOP

Pour définir un DCOP, il faut spécifier le tuple $\langle \mathcal{A}, \mathcal{X}, \mathcal{D}, \mathcal{C}, \mu, f \rangle$. Les variables de décision pour une station sont l'ensemble des décisions de servir ou non chaque utilisateur. Notons x_{ij} le nombre de blocs alloués à u_j par

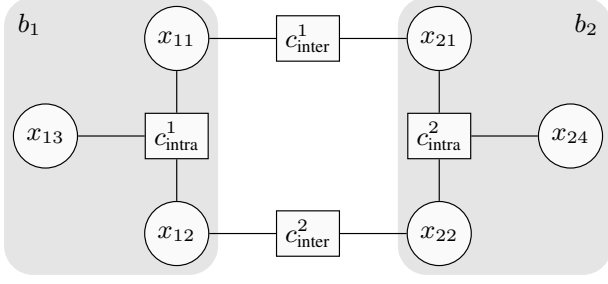


FIGURE 4 – Représentation graphique du DCOP pour le problème de la Figure 2

b_i , et donc $\mathcal{X} = \{x_{ij} \mid b_i \in \mathcal{B}_j, u_j \in \mathcal{U}\}$, où \mathcal{B}_j est l'ensemble des stations considérées par u_j , et donc nous considérons des domaines binaires $\mathcal{D}_{ij} = \{0, n_{\min}^{ij}\}$. Potentiellement, $\mathcal{B}_j = \mathcal{B}$, mais comme proposé dans [8], pour des raisons de simplification, \mathcal{B}_j peut être l'ensemble des η meilleures stations perçues par u_j du point de vue du SINR, noté \mathcal{B}_j^η . Les contraintes du DCOP, doivent assurer que le nombre de blocs utilisés sur chaque station ne dépasse pas la capacité maximale (contraintes intra-stations regroupées dans $\mathcal{C}_{\text{intra}}$), et que chaque utilisateur soit connecté à au plus une station (contraintes inter-stations regroupées dans $\mathcal{C}_{\text{inter}}$). On a l'ensemble des contraintes du DCOP $\mathcal{C} = \mathcal{C}_{\text{intra}} \cup \mathcal{C}_{\text{inter}}$. Avec ψ_i l'ensemble des utilisateurs intéressés par b_i , on a $\forall c_{\text{intra}}^j \in \mathcal{C}_{\text{intra}}$:

$$R(c_{\text{intra}}^j) = \begin{cases} -\infty & \text{si } \sum_{u_j \in \psi_i} x_{ij} > R_i \\ \sum_{u_j \in \psi_i} r_{ij} & \text{sinon} \end{cases} \quad (6)$$

et $\forall c_{\text{inter}}^i \in \mathcal{C}_{\text{inter}}$:

$$R(c_{\text{inter}}^i) = \begin{cases} 0 & \text{si } \sum_{u_j \in \psi_i} \frac{x_{ij}}{n_{\min}^{ij}} \leq 1 \\ -\infty & \text{sinon} \end{cases} \quad (7)$$

Enfin, on obtient l'équation suivante pour l'objectif global du DCOP :

$$\sum_{c \in \mathcal{C}} R(c) = \sum_{u_j \in \mathcal{U}} \sum_{b_i \in \mathcal{B}_j} r_{ij} \quad (8)$$

La fonction objectif f est donc ici la somme des taux atteints chez tous les utilisateurs, ce qui peut être considéré comme une mesure du débit dans le HetNet. Ainsi, plus $R(c)$ est grand, plus le débit devrait être bon. Enfin, le dernier choix déterminant est celui de l'ensemble des agents. Une approche intuitive est de considérer les stations comme étant des agents qui vont se coordonner pour savoir quels utilisateurs servir, car elles sont capables de rapidement communiquer entre elles, contrairement aux utilisateurs. Ainsi, ici $\mathcal{A} = \mathcal{B}$. Ce modèle est illustré en Figure 4. Cette formulation est assez simple, cependant, la présence de contraintes n-aires est une limitation assez importante pour les algorithmes DCOP par inférence, comme Max-Sum ou DPOP, qui deviennent inapplicables quand n est grand car ils reposent sur le calculs de tables de coûts de

taille exponentielle en l'arité des contraintes. En effet, les contraintes intra seront connectées à autant de variables que d'utilisateurs intéressés par une station, et dans le cas d'une station macro, elle risque de devoir gérer les demandes de tous les utilisateurs, puisqu'elle proposera un très bon débit. De plus, les contraintes inter connectent autant de variables que de stations intéressant un utilisateur donné, donc potentiellement toutes les stations. Cette limitation des contraintes inter peut cependant être contournée par la fixation du paramètre η à un nombre limité de stations candidates par utilisateur, e.g. 2 ou 3 dans [8].

3.4 À propos d'une formulation binaire

Deux techniques pour binariser des contraintes sont (i) l'ajout de variables (dites cachées) pour chaque contrainte n-aire et de contraintes binaires ou (ii) l'approche duale consistant à remplacer les contraintes par des variables (dites duales) et les variables initiales (dites primales) par des contraintes binaires [1]. Cependant, les nouvelles variables créées ont des domaines qui représentent toutes les combinaisons faisables des valeurs des n variables impliquées dans la contrainte : les valeurs de ces variables sont des vecteurs de taille n . Ainsi, la taille du domaine d'une variable duale représentant une contrainte n-aire est de taille 2^n .

Dans le cadre des contraintes sur variables binaires, une solution efficace pour stocker ces domaines de taille exponentielle consiste à utiliser les diagrammes de décision binaire (ou BDD) [5, 8], c.-à-d. représenter toutes les combinaisons possibles sous la forme d'un graphe orienté non cyclique, compression de l'arbre représentant toutes ces combinaisons. Ainsi, au lieu de stocker une très longue liste de valeurs, on stocke un automate sur lequel on peut itérer pour obtenir les valeurs du domaine. Cependant, cette technique requiert de compiler les contraintes pour générer les BDDs. Cette opération prend un temps exponentiel dans le nombre de variable dans le pire des cas, et les temps de compilation sont incompatibles avec une utilisation en cours de fonctionnement (prenant plusieurs minutes pour 30 variables, alors qu'ici nous avons jusqu'à $|\mathcal{U}| = 200$).

Ainsi, nous considérons qu'une formulation avec contraintes binaires est inenvisageable dans notre cas.

4 Analyse des performances des algorithmes DCOP

Maintenant que nous avons défini, le problème d'association des utilisateurs dans un HetNet, et sa formulations DCOP, nous allons évaluer les performances de divers algorithmes de la littérature à même de résoudre ces problèmes en un temps raisonnable.

4.1 Cadre expérimental

Nous considérons ici le HetNet à 3 tiers illustré dans la Figure 3, avec 1 station macro, 5 stations pico, et 10 stations femto, inspiré des scénarios de la littérature [8, 2]. Les puissances de transmission de ces stations sont de 40, 35 et 24 dBm, respectivement. L'environnement consiste

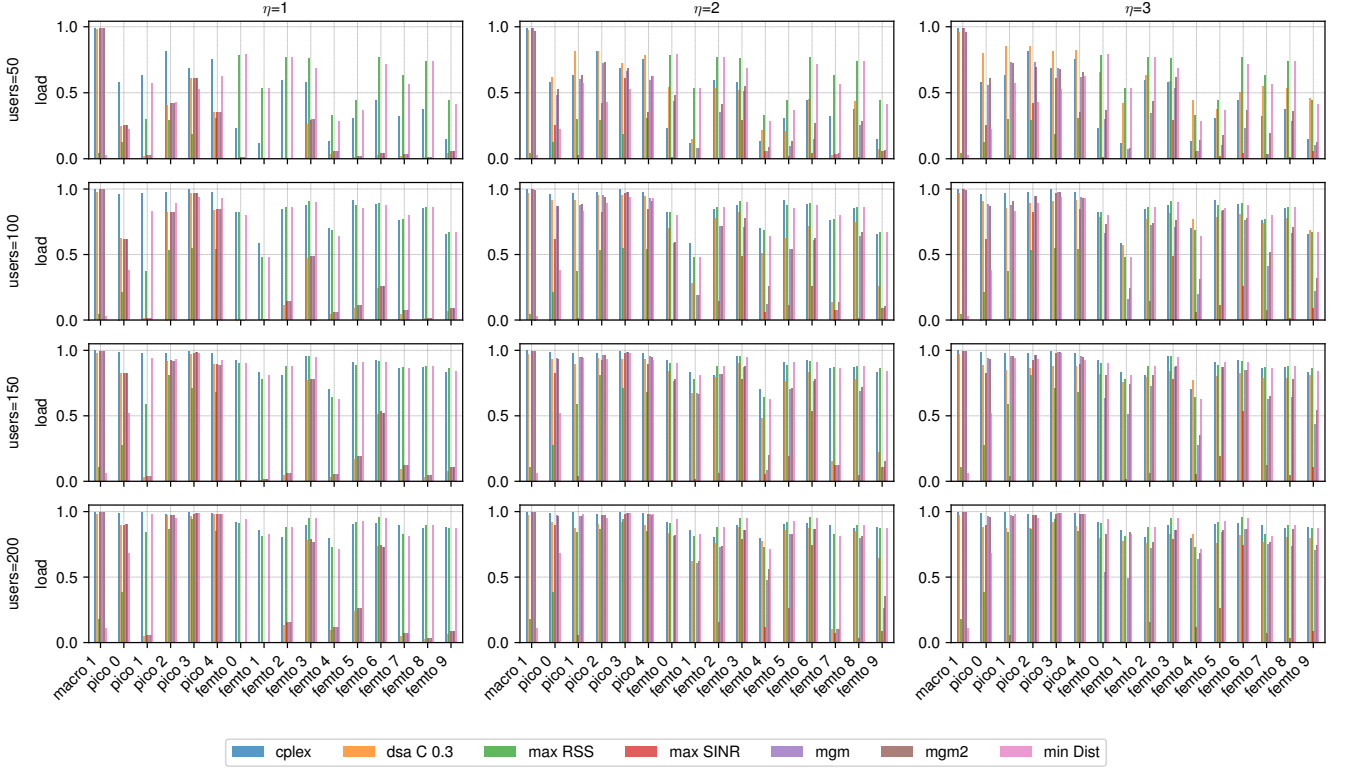


FIGURE 5 – Taux moyen d'utilisations des stations obtenus par chaque algorithme d'allocation pour différents nombres d'utilisateurs et valeurs du paramètre η

en un carré de $1000 \text{ m} \times 1000 \text{ m}$. La macro station est située au centre, et les autres stations sont positionnées aléatoirement par échantillonnage par hypercube latin [16]. Ici, nous considérerons un nombre d'utilisateurs aléatoirement positionnés souhaitant accéder au réseau dans $|\mathcal{U}| \in \{50, 100, 150, 200\}$. Avec $|\mathcal{U}| \geq 100$, le réseau est surchargé, et des utilisateurs seront non servis dans tous les cas. Le modèle de perte de puissance de signal entre une station macro ou pico et les utilisateurs est défini comme $L(d) = 34 + 40 \log_{10} d$, avec d la distance en mètres entre la station et l'utilisateur. De même, pour les stations femto avec $L(d) = 37 + 30 \log_{10} d$. La puissance du bruit est -111.45 dBm , qui est le bruit thermique à la température dans une pièce, avec une largeur de bande de 180 kHz . L'intervalle de temps de l'équation (3) est $t = 1$ seconde et $\Gamma = 1$. Le nombre de blocs disponibles est de 100 pour les stations macro, 50 pour les stations pico, et 25 pour les stations femto. Nous considérons $\eta \in \{1, 2, 3\}$.

Les algorithmes suivants sont évalués : (i) allocation optimale, avec CPLEX, en formulant le DCOP n-aire sous forme de programme linéaire en nombres entiers ; (ii) allocation par maxSINR, ce qui revient pour chaque utilisateur à s'associer à la station disponible ayant le maximum de SINR (en cas de non disponibilité, l'utilisateur passe à un second choix, etc.) ; (iii) allocation par maxRSS, équivalent à maxSINR, mais avec la puissance sans interférence ni bruit comme critère de choix ; (iv) minDist, équivalent à maxSINR, mais avec la distance comme critère de choix ; (v) MGM ; (vi) MGM-2 ; (vii) DSA-C (probabilité fixée à

0.3)¹. Nous avons réduit l'étude à ces algorithmes car les algorithmes DCOP optimaux, n'ont pas fourni de résultats en un temps acceptable (timeout de 60s). Pour chaque valeur de η et de N , nous générons 20 instances de DCOP, et chaque instance est résolue 10 fois par chaque algorithme DCOP, partant de l'allocation obtenue par maxSINR. Les autres algorithmes étant déterministes, ils ne sont appliqués qu'une seule fois. Nous affichons la moyenne et l'écart-type (parfois très faible donc peu visible) des mesures sur ces 20 ou 200 instances. MGM, MGM-2 et DSA ont été paramétrés pour fournir des solutions au bout de 20 itérations.

Les calculs ont été effectués sur une machine avec processeur de type Intel(R) Xeon(R) CPU E5-1603 v3 @ 2.80 GHz et 32 GiB de mémoire, en utilisant la librairie pyDCOP [22, 21], des codes écrits en Python 3.8, excepté pour les solutions optimales calculées avec CPLEX version 12.61 [6] en utilisant l'API Python via la librairie PuLP [20].

4.2 Analyse des performances

La Figure 5 présente la charge moyenne de chaque station de base pour chaque configuration et chaque algorithme. Comme précisé dans la littérature, maxSINR implique une forte charge sur la station macro, puis, lorsque le nombre d'utilisateurs augmente, la charge se déporte sur les stations de tiers inférieurs. Tous les algorithmes pré-

1. Cette configuration de DSA a été celle présentant les meilleurs résultats parmi la famille DSA sur ce problème

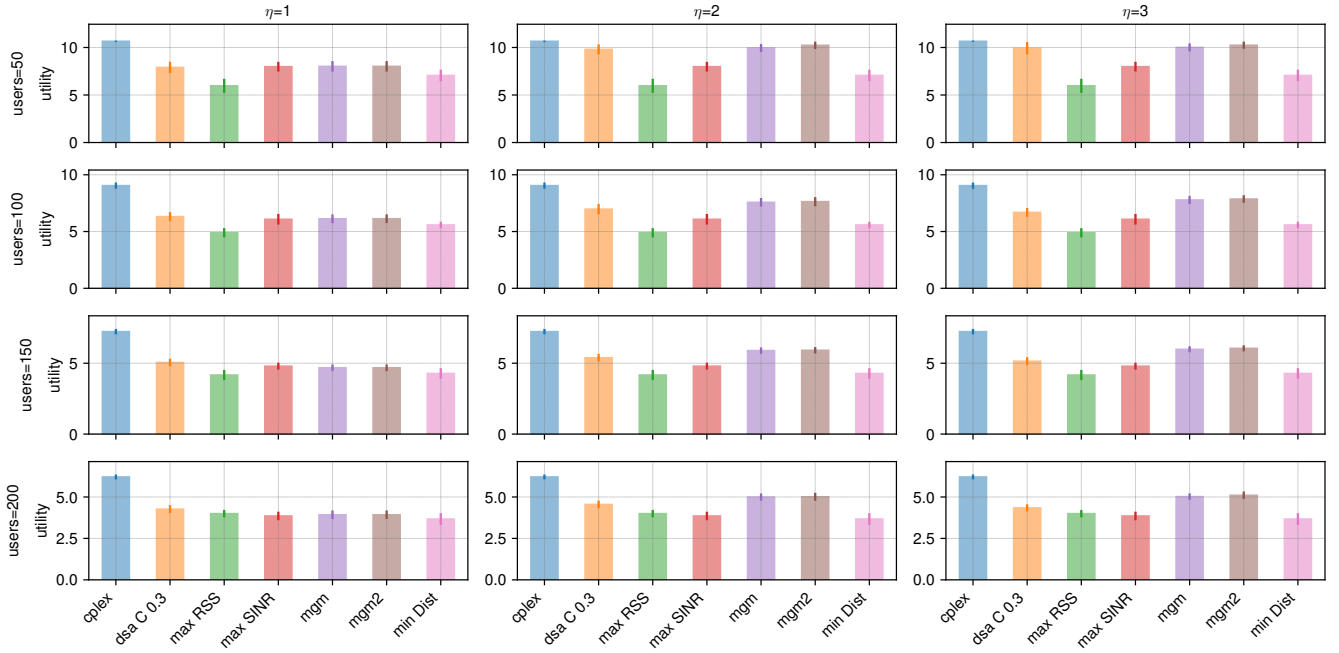


FIGURE 6 – Utilité finale (débit moyen par utilisateur en Mbps) obtenue par chaque algorithme d'allocation pour différents nombres d'utilisateurs et valeurs du paramètre η

sentent ce comportement, excepté maxRSS et minDist qui vont principalement affecter les utilisateurs aux stations les plus proches, et comme les utilisateurs sont répartis sur tout l'environnement, toutes les stations ont une charge conséquente. Pour $\eta = 1$, notons que les algorithmes DCOP ont le même comportement que maxSINR, et vont donc mettre de côté les stations de tiers inférieurs, puisque les utilisateurs ne considèrent que la meilleure station du point de vue du SINR. Avec $\eta > 1$, les stations partent de l'allocation maxSINR, puis se coordonnent avec un algorithme DCOP pour améliorer la solution. Nous obtenons alors des allocations différentes à maxSINR, proches des allocations optimales calculées par CPLEX, qui sont de meilleure qualité du point de vue des utilisateurs.

En effet, en analysant l'utilité globale du système, c.-à-d. le débit global en Mbps, présenté dans la Figure 6, nous pouvons observer que pour $\eta = 1$, les algorithmes DCOP obtiennent la même qualité que maxSINR (puisque pas de réelle coordination), à part DSA. Ce dernier, grâce à ses mouvements aléatoires, change l'allocation maxSINR initiale, et donc retire des utilisateurs de certaines stations, et ajoutent des utilisateurs initialement non servis, pour obtenir en moyenne des solutions de meilleure qualité. Cependant, cette qualité n'est pas réellement opérationnelle, puisque DSA viole de nombreuses contraintes intra, et donc surcharge des stations au-delà de leur limite. Pour $\eta > 1$, la qualité des solutions MGM et MGM-2 augmente, et se place entre la qualité optimale et la qualité de maxSINR. Avec η augmentant, MGM-2 fournit des qualités de solutions très légèrement supérieures à celles fournies par MGM. Les pires solutions sont obtenues par maxRSS et minDist.

Cette qualité supérieure s'explique ainsi par l'ajout de co-

ordination entre les stations, qui permet de servir plus d'utilisateurs, comme illustré dans la Figure 7. On y retrouve la même performance pour MGM-2 avec $\eta = 3$, qui se positionne entre la solution maxSINR et la solution optimale théorique obtenue par CPLEX. Notons que les instances que nous avons générées sont très stressantes pour le système, puisque tous les utilisateurs ne peuvent être physiquement servis à partir de $|\mathcal{U}| \gtrsim 85$. Sur les systèmes non surchargés ($|\mathcal{U}| = 50$), MGM et MGM-2 améliorent grandement la part d'utilisateurs servis. Cet écart diminue dans les systèmes surchargés, mais MGM-2 fournit toujours de meilleures solutions.

Cependant cette coordination nécessite une charge réseau conséquente pour la stations, afin de traiter les messages de coordination, comme illustré en Figure 8, mais notons que ces messages DCOP ne se propagent qu'entre stations de base (donc à haut débit, et faible latence), et surtout que ces protocoles sont peu sensibles à la perte de messages, contrairement à une solution qui consisterait à recueillir toutes les données sur les utilisateurs (position, signal, SINR, exigences, etc.) et sur les stations de base (charge), puis les faire remonter à un serveur global pour effectuer une optimisation globale, avec CPLEX, par exemple.

Au final, MGM, pour une quantité d'information moindre que MGM-2, permet toutefois d'obtenir des solutions de bonne qualité, si une surcharge de messages entre stations est problématique.

5 Conclusions

Dans cet article, nous avons étudié les performances d'algorithmes DCOP légers pour l'association d'utilisateurs à des réseaux HetNet. Ici, l'objectif étant une optimisa-

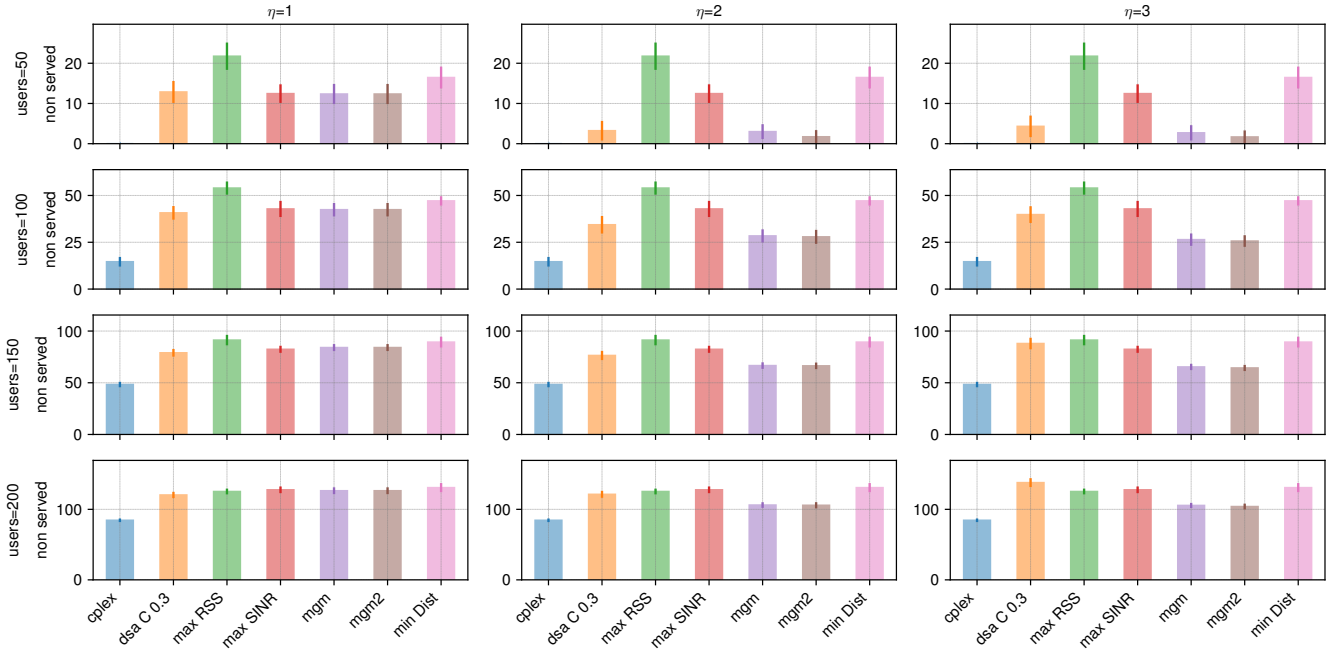


FIGURE 7 – Nombre final d'utilisateurs sans connexion obtenu par chaque algorithme d'allocation pour différents nombres d'utilisateurs et valeurs du paramètre η

tion du débit global obtenu par les utilisateurs, nous avons comparé les algorithmes MGM, MGM-2, et DSA-C à des algorithmes d'allocation classiques dans le domaines (maxSINR, maxRSS) ainsi qu'à la solution optimale théorique obtenue de manière centralisée par programmation mathématique (CPLEX). Dans des environnements chargés, MGM et MGM-2 exhibent des performances se situant entre la solution distribuée classiquement utilisée, maxSINR, et l'optimum théorique. Ainsi, en ajoutant une couche de coordination par DCOP entre les stations de base, le système améliore les performances normalement obtenues dans un HetNet. Cette amélioration requiert uniquement un échange de messages entre les stations (et donc à haut débit et sans goulot d'étranglement).

Cette étude aboutit ainsi sur des résultats prometteurs quant à l'utilisation des techniques DCOP dans le cadre des HetNet. Cependant, nous n'avons utilisé que des algorithmes génériques issus de la littérature. Afin d'améliorer encore les performances, il nous semble nécessaire d'adapter ces algorithmes et leurs structures de données au cas particulier des HetNets, notamment à cause de la présence de contraintes n -aires et de la mixité contraintes souples/contraintes dures. Enfin, l'un des apports envisagés d'une approche décentralisée est la capacité à adapter le système en local, en cas de défaillance d'équipements. Nous envisageons ainsi de poursuivre cette étude dans un cadre DCOP dynamique, en utilisant notamment les mécanismes de résilience proposés dans [23].

Références

- [1] F. Bacchus and P. van Beek. On the conversion between non-binary constraint satisfaction problems. In

Proceedings of the Fifteenth National/Tenth Conference on Artificial Intelligence/Innovative Applications of Artificial Intelligence, AAAI '98/IAAI '98, page 311–318, USA, 1998. American Association for Artificial Intelligence.

- [2] B. Bikram Kumar, L. Sharma, and S.-L. Wu. Online distributed user association for heterogeneous radio access network. *Sensors*, 19(6), 2019.
- [3] H. Boostanimehr and V. K. Bhargava. Unified and distributed qos-driven cell association algorithms in heterogeneous networks. *IEEE Transactions on Wireless Communications*, 14(3) :1650–1662, March 2015.
- [4] S. Boudko, W. Leister, and S. Gjessing. Heterogeneous wireless network selection : Load balancing and multicast scenario. *International Journal on Advances in Networks and Services*, 6 :118–135, 12 2013.
- [5] R.E. Bryant. Graph-based algorithms for boolean function manipulation. *IEEE Transactions on Computers*, C-35(8) :677–691, 1986.
- [6] <https://www.ibm.com/analytics/cplex-optimizer>.
- [7] A. Damnjanovic, J. Montojo, Y. Wei, T. Ji, T. Luo, M. Vajapeyam, T. Yoo, O. Song, and D. Malladi. A survey on 3gpp heterogeneous networks. *IEEE Wireless Communications*, 18(3) :10–21, June 2011.
- [8] P. Duan, C. Zhang, G. Mao, and B. Zhang. Applying distributed constraint optimization approach to the user association problem in heterogeneous networks. *IEEE Transactions on Cybernetics*, 48(6) :1696–1707, 2018.

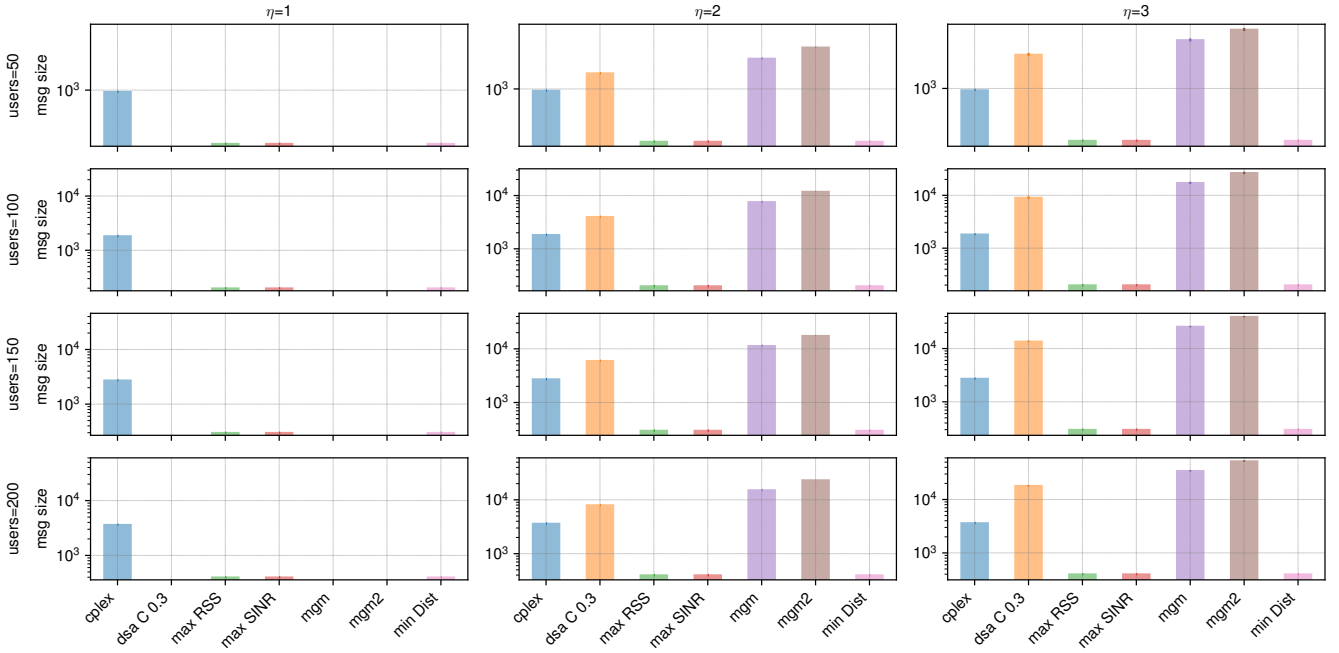


FIGURE 8 – Charge réseau induite par chaque algorithme d'allocation pour différents nombres d'utilisateurs et valeurs du paramètre η

- [9] A. Farinelli, A. Rogers, and N. R. Jennings. Agent-based decentralised coordination for sensor networks using the max-sum algorithm. *Autonomous Agents and Multi-Agent Systems*, 28(3) :337–380, May 2014.
- [10] F. Fioretto, E. Pontelli, and W. Yeoh. Distributed constraint optimization problems and applications : A survey. *Journal of Artificial Intelligence Research*, 61 :623–698, 2018.
- [11] D. Fooladivanda and C. Rosenberg. Joint resource allocation and user association for heterogeneous wireless cellular networks. *IEEE Transactions on Wireless Communications*, 12(1) :248–257, 2013.
- [12] K. Gomadam, V. R. Cadambe, and S. A. Jafar. A distributed numerical approach to interference alignment and applications to wireless interference networks. *IEEE Transactions on Information Theory*, 57(6) :3309–3322, June 2011.
- [13] V. N. Ha and L. B. Le. Distributed base station association and power control for heterogeneous cellular networks. *IEEE Transactions on Vehicular Technology*, 63(1) :282–296, Jan 2014.
- [14] M. Jain, M. Taylor, M. Tambe, and M. Yokoo. DCOPs meet the real world : Exploring unknown reward matrices with applications to mobile sensor networks. In *International Joint Conference on Artificial Intelligence (IJCAI'09)*, pages 181–186, 2009.
- [15] R.T. Maheswaran, J.P. Pearce, and M. Tambe. Distributed algorithms for DCOP : A graphical-game-based approach. In *Proceedings of the 17th International Conference on Parallel and Distributed Computing Systems (PDCS)*, pages 432–439, 2004.
- [16] M. D. McKay, R. J. Beckman, and W. J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 42(1) :55–61, 2000.
- [17] P. J. Modi, W. Shen, M. Tambe, and M. Yokoo. ADOPT : Asynchronous Distributed Constraint Optimization with Quality Guarantees. *Artificial Intelligence*, 161(2) :149–180, 2005.
- [18] A. Petcu and B. Faltings. A scalable method for multiagent constraint optimization. In *International Joint Conference on Artificial Intelligence (IJCAI'05)*, pages 266–271, 2005.
- [19] Gauthier Picard. Optimisation sous contraintes distribuée : une introduction au domaine. In *Journées Francophones sur les Systèmes Multi-Agents*, pages 43–52. Cépaduès, 2018.
- [20] <https://pythonhosted.org/PuLP/>.
- [21] <https://github.com/Orange-OpenSource/pyDcop>.
- [22] P. Rust, G. Picard, and F. Ramparany. pyDCOP, a DCOP library for IoT and dynamic systems. In *International Workshop on Optimisation in Multi-Agent Systems (OptMAS@AAMAS 2019)*, 2019.
- [23] P. Rust, G. Picard, and F. Ramparany. Resilient distributed constraint optimization in physical multi-agent systems. In *European Conference on Artificial Intelligence (ECAI)*. IOS Press, 2020.
- [24] Z. Wang, X. Zhu, X. Bao, and S. Zhao. A novel resource allocation method in ultra-dense network based on noncooperation game theory. *China Communications*, 13(10) :169–180, Oct 2016.

- [25] W. Zhang, G. Wang, Z. Xing, and L. Wittenburg. Distributed stochastic search and distributed breakout : Properties, comparison and applications to constraint optimization problems in sensor networks. *Artificial Intelligence*, 161(1-2) :55–87, January 2005.
- [26] L. Zhao, Y. Qin, M. Ma, X. Zhong, and L. Li. Qos guaranteed resource block allocation algorithm in lte downlink. In *7th International Conference on Communications and Networking in China*, pages 425–429, Aug 2012.