ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY

TELECOM ParisTech

2017-ENST-0030

EDITE - ED 130

Doctorat ParisTech

T H È S E

pour obtenir le grade de docteur délivré par

TELECOM ParisTech

Spécialité « Informatique et Réseaux »

présentée et soutenue publiquement par

Mohammed Shabbir ALI

27 Juin 2017

Titre

Apprentissage distribué dans les jeux pour les réseaux sans fil

Directeur de thèse : **Marceau COUPECHOUX**
Co-encadrement de la thèse : **Pierre COUCHENEY**

Jury

M. Bruno GAUJAL, Directeur de Recherches, INRIA	Rapporteur
M. Ekram HOSSAIN, Professeur, University of Manitoba	Rapporteur
M. Eitan ALTMAN, Directeur de Recherches, INRIA	Examinateur
Mme. Johanne COHEN, Directeur de Recherches, CNRS	Examinateur
M. Jean-Marie GORCE, Professeur, INSA-Lyon	Examinateur
M. Olivier MARCÉ, Ingénieur de Recherche, Nokia Bell Labs	Examinateur

TELECOM ParisTech
école de l'Institut Mines-Télécom - membre de ParisTech
46 rue Barrault 75013 Paris - (+33) 1 45 81 77 77 - www.telecom-paristech.fr

T H È S E

Abstract

Dans cette thèse, nous abordons la question de la conception et de la gestion optimales des réseaux sans fil, qui sont devenus essentiels avec la demande croissante de données. Les approches centralisées deviennent impraticables, tandis que l'apprentissage distribué dans les jeux est une approche distribuée prometteuse. En particulier, les jeux potentiels sont attrayants car les joueurs du jeu peuvent optimiser de façon distribuée une fonction potentielle. Nous étendons cette notion aux jeux potentiels approchés et bruités et étudions la convergence des algorithmes d'apprentissage dans ces cadres. Nous appliquons ensuite les résultats théoriques obtenus aux réseaux sans fil. Dans une première partie, nous prouvons que, dans certaines conditions, l'algorithme d'apprentissage log-linéaire (LLA) et le LLA binaire (BLLA) convergent vers le minimum global de la fonction potentielle dans les jeux potentiels approchés. Nous prouvons alors la convergence de BLLA dans des jeux potentiels bruités pour des températures fixes et décroissantes. Nous fournissons un nombre suffisant d'échantillons d'estimation qui garantit la convergence pour un bruit borné ou non borné. Un facteur clé pour analyser la convergence des algorithmes proposés est la résistance des arbres dans une Chaîne de Markov Perturbée (PMC). Nous développons de nouvelles règles pour simplifier le calcul de la résistance des arbres. Dans une deuxième partie, nous appliquons nos résultats à des problèmes pratiques dans les réseaux sans fil. Nous considérons d'abord l'équilibrage de charge dans des réseaux cellulaires hétérogènes. Dans ce contexte, nous proposons également un nouveau schéma de recuit pour LLA adapté aux horizons finis. Ensuite, nous appliquons BLLA au problème d'assignation de canal (CAP) dans les réseaux de communication de terminal à terminal (D2D). En raison du bruit d'estimation des débits, le cadre de jeu à potentiel bruité se pose naturellement. Nous abordons enfin un problème de maximisation de fonction sous modulaire avec contraintes. Nous le concevons comme un jeu fini avec des joueurs ayant des informations limitées ou inexistantes sur leur voisinage. Nous caractérisons la performance de l'algorithme glouton et nous fournissons le meilleur graphe d'information pour un nombre donné de joueurs et d'arêtes d'information.

Publications

Journal Papers

J1 Mohd. Shabbir Ali, Pierre Coucheney, and Marceau Coupechoux "*Load Balancing in Heterogeneous Networks Based on Distributed Learning in Near-Potential Games*", *IEEE Trans. Wireless Commun.*, vol.15, no.7, pp.5046-5059, 2016.

J2 Mohd. Shabbir Ali and Neelesh B. Mehta, "*Modeling Time-Varying Aggregate Interference in Cognitive Radio Systems, and Application to Primary Exclusive Zone Design*," *IEEE Trans. Wireless Commun.*, vol.13, no.1, pp.429-439, Jan. 2014.

Conference Papers

C1 David Grimsman, Mohd. Shabbir Ali, João P. Hespanha and Jason R. Marde " *Efficiency and Information Trade-off of Submodular Maximization Problems*", Accepted in *IEEE Conference on Decision and Control*, 2017.

C2 Mohd. Shabbir Ali, Pierre Coucheney, and Marceau Coupechoux " *Optimal distributed channel assignment in D2D networks using learning in noisy potential games*", *Accepted in INFOCOM 5G and Beyond Workshop*, May 2017.

C3 Mohd. Shabbir Ali, Pierre Coucheney, and Marceau Coupechoux " *Rules for Computing Resistance of Transitions of Learning Algorithms in Games*", Accepted in *International Conference on Game Theory for Networks (GAMENETS)*, May 2017.

C4 Mohd. Shabbir Ali, Pierre Coucheney, and Marceau Coupechoux "*Learning Annealing Schedule of Log-Linear Algorithms for Load Balancing in HetNets*", *Proc. European Wireless Conference* pp. 1-6, 2016.

C5 Mohd. Shabbir Ali, Pierre Coucheney, and Marceau Coupechoux "*Load Balancing in Heterogeneous Networks Based on Distributed Learning in Potential Games*", Proc. *IEEE WiOpt,* pp.371-378, May 2015.

C6 Mohd. Shabbir Ali and N. B. Mehta, "*Modeling time-varying aggregate interference from cognitive radios and implications on primary exclusive zone design*", *IEEE Global Communications Conference (GLOBECOM)*, pp. 3760-3765, 2013.

arXiv Paper

X1 Mohd. Shabbir Ali, Pierre Coucheney, and Marceau Coupechoux " *Optimal distributed channel assignment in D2D networks using learning in noisy potential games*", *arXiv preprint arXiv:1701.04577*, 2017.

Contents

List of Tables

List of Figures

Chapter 1

Apprentissage distribué dans les jeux pour les réseaux sans Fil

1.1 Introduction

La durabilité du monde dépend des systèmes d'ingénierie de contrôle tels que les systèmes de réseaux de communication, les systèmes d'alimentation électrique, les systèmes de transport, etc., qui répondent à ses besoins. La taille de ces systèmes augmente de façon exponentielle avec les exigences toujours croissantes de la population. La conception et la gestion efficaces de ces systèmes sont essentielles. Les approches centralisées à cet effet sont devenues impraticables en raison de leur grande taille et de leur grande complexité. Les approches distribuées semblent être les seules alternatives prometteuses. Le défi consiste à concevoir des algorithmes distribués qui peuvent atteindre les performances globales optimales.

La théorie des jeux est attrayante puisqu'elle fournit un outil d'approche distribuée. De nombreux problèmes d'ingénierie peuvent être modélisés en tant que jeu. Un jeu consiste en un ensemble de joueurs ou d'agents, de leurs actions et de leurs fonctions d'utilité ou de coût. Les multiples joueurs interagissent les uns avec les autres pour arriver à un état d'équilibre. Une notion d'équilibre est celui de Nash (NE), dans lequel aucun joueur ne peut gagner par écart unilatéral. Un défi important est de savoir comment atteindre le NE optimal, celui qui correspond à l'optimum global de la fonction objectif. Les jeux potentiels sont prometteurs pour relever ce défi. En effet, dans ces jeux, les joueurs peuvent atteindre l'optimum global de manière distribuée. Les jeux potentiels approchés élargissent la portée des jeux potentiels à une classe plus large de problèmes. Ils relâchent les contraintes

strictes des jeux potentiels et, en même temps, partagent des propriétés similaires de nature distribuée. Les jeux potentiels bruités élargissent encore la portée des jeux potentiels à une classe de problèmes d'optimisation stochastique en tenant compte des utilités bruités des joueurs. Cependant, pour obtenir une structure de jeux potentiels, les utilités ou les fonctions de coût des joueurs doivent être soigneusement conçus, ce qui n'est peut-être pas toujours possible. Cela peut être dû au manque d'informations chez les joueurs . Cependant, si la fonction objectif possède des propriétés telles que la propriété sous modulaire, elle peut être exploitée.

Dans l'apprentissage distribué, les joueurs apprennent de façon distribuée leurs actions optimales. Seuls les informations locales sur le voisinage peuvent être disponibles pour les joueurs. Cela permet de résoudre efficacement les problèmes sans trop de surcharge en termes d'échanges d'informations. Par conséquent, l'apprentissage distribué trouve des applications dans de nombreux systèmes d'ingénierie tels que les réseaux de communication sans fil, les réseaux de communication de terminal à terminal (D2D) et les réseaux de capteurs sans fil. L'apprentissage distribué peut être appliqué à des problèmes d'optimisation qui peuvent être modélisés comme des jeux potentiels. À cette fin, la fonction objectif du problème est transformée en fonction potentielle du jeu. L'objectif optimal de l'objectif sera donc le NE optimal du jeu. Pour une classe différente de fonctions potentielles, différents algorithmes d'apprentissage distribués sont nécessaires. Pour la classe de fonctions de potentiel convexe ou concave, un simple algorithme glouton ou de meilleure réponse peut atteindre le NE optimal. Cependant, en général, la convexité ne se produit pas dans de nombreux problèmes. L'algorithme de meilleure réponse ne garantit pas l'optimum car il peut se trouver piégé dans un minimum local de la fonction non-convexe. Ici, le défi consiste à concevoir de nouveaux algorithmes distribués qui peuvent échapper au minimum local et atteindre le minimum global. En outre, si la fonction potentielle est corrompue par une sorte de bruit comme le bruit d'estimation, le problème devient encore plus difficile.

En résumé, l'apprentissage distribué peut être appliqué à une grande variété de problèmes où les approches centralisées deviennent impraticables. L'objectif de la thèse est d'étudier les algorithmes d'apprentissage distribués dans les jeux. Ensuite, les algorithmes sont appliqués pour résoudre les problèmes des réseaux de communication sans fil. Dans ce qui suit, nous présentons d'abord les principales contributions de la thèse puis nous donnons un aperçu de la thèse. De plus, une documentation connexe sur les approches d'apprentissage distribué est présentée.

1.1.1 Principales contributions de la thèse

Nous résumons les principales contributions de la thèse ci-dessous.

1. *Algorithmes d'apprentissage distribués dans les jeux potentiels approchés :*
   Nous étudions deux algorithmes d'apprentissage distribués pour les jeux potentiels approchés dans la section 1.2 : d'abord, l'algorithme d'apprentissage log-linéaire (LLA) ; deuxièmement, un algorithme d'apprentissage logarithmique binaire (BLLA) à information partielle dans lequel les joueurs ont des informations seulement sur deux de leurs actions. L'idée clé de ces deux algorithmes est parfois de jouer des actions sous-optimales pour échapper à un minimum local et atteindre le minimum global. La probabilité de jouer des actions sous-optimales est régie par un paramètre de température τ. Nous prouvons que sous certaines conditions, LLA et BLLA convergent vers le minimum global de la fonction potentielle.

2. *Algorithme d'apprentissage réparti dans les jeux potentiels bruités :*
   Nous présentons les jeux potentiels bruités dans la section 1.2, pour tenir compte du bruit dans les utilités des joueurs. L'objectif est d'atteindre la valeur moyenne optimale de la fonction potentielle. Les jeux potentiels bruités peuvent être utilisés pour modéliser le bruit d'une classe de problèmes d'optimisation stochastiques (SOP) dont la fonction objectif comporte une composante de bruit. Ensuite, nous étudions un algorithme d'apprentissage logarithmique binaire distribué (BLLA) pour les jeux potentiels bruités. Dans ce contexte, les joueurs jouent plusieurs fois leurs actions pour diminuer la variance du bruit afin d'obtenir la convergence. Le nombre de fois où une action est jouée dépend du fait que le bruit est borné ou non. La convergence de BLLA dans un jeu potentiel bruité est prouvée pour une température fixe et une température décroissante. Nous fournissons un nombre suffisant d'échantillons d'estimation qui garantisse la convergence pour les cas de bruit borné et de bruit non borné.

3. *Règles de résistance pour analyser les algorithmes d'apprentissage :*
   Nous présentons des règles pour les calculs de la résistance des arbres dans une Chaîne de Markov Perturbé (PMC) dans la section 1.2.4. La résistance des arbres est un facteur clé pour analyser la convergence des algorithmes proposés. Intuitivement, la résistance d'une arête d'un arbre est le coût d'une action sous-optimale. Les algorithmes proposés induisent une chaîne de Markov perturbée qui peut être décrite par une fonction de transition de probabilité (TPF). En utilisant la TPF de l'algorithme,

la résistance peut être calculée. Cependant, la TPF peut être composite et complexe, ce qui peut rendre très difficile le calcul de la résistance. Par conséquent, nous développons des règles pour simplifier le calcul de la résistance des arbres pour les TPF complexes et composites. Ces règles transforment le calcul de la résistance des fonctions composites en un calcul de résistance des fonctions élémentaires simples. Ainsi, une analyse facile des algorithmes d'apprentissage proposés est possible. En particulier, nous verrons que ces règles sont essentielles dans la preuve de convergence de BLLA dans les jeux potentiels bruités.

4. *Equilibrage de la charge dans les réseaux hétérogènes :*
   Dans la section 1.3, nous appliquons les algorithmes d'apprentissage proposés dans les jeux potentiels approchés pour l'équilibrage de charge dans des réseaux hétérogènes qui utilisent les biais (notés CRE) pour l'association des utilisateurs et la technique de réduction d'interférence dite ABS. Nous avons d'abord modélisé le problème de l'équilibrage de charge en minimisant de manière contrainte une fonction objectif d'$\alpha-$équité avec des contraintes de charge et de dépassement. La fonction objectif d'$\alpha-$équité permet d'atteindre diverses performances et équités du réseau pour différents α. Une politique optimale de débit est obtenue pour $\alpha = 0$, l'équité proportionnelle pour $\alpha = 1$, le délai moyen minimum pour $\alpha = 2$ et la politique de charge min-max quand $\alpha \to \infty$. Nous fournissons une preuve détaillée de la politique de charge min-max pour $\alpha \to \infty$. Cette preuve étend le résultat classique dérivé de [1] en considérant une fonction d'$\alpha-$équité non convexe. Ensuite, nous modélisons le problème de minimisation sous contrainte ci-dessus grâce à un jeu potentiel approché dans lequel les stations de base sont les joueurs et leurs actions est un ensemble de biais CRE et de rapports ABS. Nous avons transformé la fonction objectif en fonction potentielle en concevant les fonctions de coûts des stations de base en fonction de leur voisinage. En présence d'effet de masque, le voisinage des stations de base peut être important et inclure théoriquement tout le réseau. Nous résolvons le problème de façon distribuée en construisant de petits voisinages approximatifs des stations de base. Pour ce faire, nous proposons également une technique pour construire itérativement les voisinages qui permettent l'utilisation des fonctions de coûts. Ensuite, le minimum global de la fonction objectif est atteint en utilisant le LLA et le BLLA proposés dans les jeux potentiels approchés. Une contribution importante est de dériver des conditions pratiques en termes de paramètres système selon lesquels BLLA et LLA convergent vers le minimum global. En exécutant

des simulations, nous montrons que les algorithmes proposés convergent vers le minimum global.

5. *Algorithme d'apprentissage de recuit pour LLA et BLLA :*
   Nous proposons un nouvel algorithme d'apprentissage de recuit (ALA) pour apprendre le paramètre de température τ de LLA et BLLA. ALA est développé en adaptant l'algorithme de rejet successif [2]. Dans ALA, un large ensemble de valeurs possibles de τ et un horizon temporel sont considérés. La meilleure valeur τ parmi un ensemble de valeurs est automatiquement obtenue par ALA en évaluant les algorithmes pour l'ensemble des valeurs. Dans l'évolution de LLA, ALA vérifie la performance de différents τ pour rejeter successivement ceux avec la pire performance. À la fin de l'horizon, ALA sélectionne le τ le plus performant. Le processus consistant à rejeter successivement l'ensemble des résultats indésirables de τ constitue un schéma de recuit pour LLA et BLLA. Nous montrons que ce schéma de recuit est rapide, fonctionne mieux que d'autres et garantit une convergence asymptotique.

6. *Affectation de canal dans les réseaux D2D :*
   Dans la section 1.4, nous appliquons les algorithmes d'apprentissage proposés dans les jeux potentiels bruités au problème d'affectation de canal dans les réseaux sans fil D2D. Tout d'abord, le CAP est modélisé comme un SOP, où l'objectif est d'obtenir la somme des débits du réseau maximale. Ensuite, le SOP est traduit en un jeu potentiel bruité qui prend en compte le bruit d'estimation des débits de données. Nous adoptons le BLLA distribué dans le jeu potentiel bruité pour atteindre les assignations de canal optimales. De nombreuses simulations montrent que le BLLA proposé atteint la somme des débit maximale.

7. *Algorithme glouton distribué dans les jeux avec fonction objectif sous-modulaire :*
   Dans la section 1.5, nous d'autres jeux que les jeux potentiels. Nous considérons un système où la fonction objectif est sous-modulaire. Nous considérons un problème généralisé de maximisation sous-modulaire avec contrainte. Nous le concevons comme un jeu fini avec des joueurs ayant des informations limitées ou inexistantes sur leur voisinage. L'information du voisinage est représentée sous la forme d'un diagramme acyclique dirigé (DAG), que nous appelons graphe d'information. Nous étudions la performance d'un algorithme glouton distribué pour les joueurs avec des informations limitées. Nous donnons d'abord des limites inférieures et supérieures aux performances les plus défavorables de l'algorithme glouton pour un graphe

d'information donné. Les limites donnent une intuition sur les propriétés du graphe sous-jacent qui doivent être améliorées afin que les performances puissent s'améliorer. Cela permet de déterminer le meilleur graphe d'information qui donne les meilleures performances qu'un concepteur de système peut réaliser avec un nombre fixe d'agents et d'arêtes d'informations. Ces résultats montrent que lorsque les informations sont coûteuses, les meilleures structures de communication répartissent les liens de communication entre les agents, plutôt que de les regrouper dans un petit groupe.

1.2 Apprentissage distribué dans les jeux potentiels

Dans cette section, nous présentons l'apprentissage dans des jeux potentiels pour résoudre de manière distribuée les problèmes d'optimisation.

Nous considérons un jeu fini $\Gamma = \left\{\mathscr{S}, \{X_i\}_{i\in\mathscr{S}}, \{U_i\}_{i\in\mathscr{S}}\right\}$, où $\mathscr{S}$ est l'ensemble de joueurs, $X = X_1 \times X_2 \times \ldots \times X_{|\mathscr{S}|}$ est un ensemble de stratégies ou un ensemble d'actions, et $U_i : X \to \mathscr{R}$ est une fonction d'utilité ou de coût. Différents jeux peuvent être obtenus en concevant de manière variée les fonctions de coûts. Un profil de stratégie du jeu est désigné par $x = (x_i, x_{-i})$, où x_i est la stratégie du joueur i et x_{-i} est le profil de stratégie tous les joueurs sauf le joueur i. Les joueurs jouent à plusieurs reprises leurs actions pour arriver à un point d'équilibre s'il existe. Dans un équilibre approché de Nash ou $\varepsilon-$NE, aucun joueur ne peut bénéficier de plus de ε en changeant unilatéralement sa stratégie. Formellement, $\varepsilon-$NE est défini comme ci-dessous.

Definition 1 *[$\varepsilon-$Nash Equilibrium:] Un profil de stratégie $\left(x_i^*, x_{-i}^*\right) \in X$ est un ε-NE si*

$$U_i(x_i^*, x_{-i}^*) - U_i(x_i, x_{-i}^*) \leq \varepsilon, \quad \forall x_i \in X_i, \forall i \in \mathscr{S}. \tag{1.1}$$

Si $\varepsilon = 0$, il s'agit d'un NE pur (PNE). Si les fonctions d'utilité sont considérées à la place des fonctions de coût des joueurs, le signe d'inégalité ci-dessus change de $\leq$ à $\geq$.

1.2.1 Jeux potentiels exacts

Definition 2 *[jeu potentiel exact [3]] $\Gamma = \left\{\mathscr{S}, \{X_i\}_{i\in\mathscr{S}}, \{U_i\}_{i\in\mathscr{S}}\right\}$ est un jeu de potentiel exact s'il existe une fonction appelée fonction potentielle $\Phi : X \to \mathscr{R}$ telle que $\forall i \in \mathscr{S}$, $\forall x_i, x_i' \in X_i$ et $\forall x_{-i} \in X_{-i}$,*

$$U_i(x_i, x_{-i}) - U_i(x_i', x_{-i}) = \Phi(x_i, x_{-i}) - \Phi(x_i', x_{-i}). \tag{1.2}$$

Un jeu potentiel exact a au moins un PNE et les optimiseurs locaux de la fonction potentielle sont les PNE [3]. Un PNE optimal est un profil d'action x^* qui correspond à l'optimum global de la fonction potentielle.

Sur la base de la notion développée dans [4], nous définissons maintenant un jeu potentiel approché ou ξ-potentiel comme ci-dessous.

Definition 3 *[ξ-potential game] Un jeu $\mathscr{G} = \left\{\mathscr{S}, \{X_i\}_{i\in\mathscr{S}}, \{U_i\}_{i\in\mathscr{S}}\right\}$ est un jeu xi -potentiel s'il existe une fonction potentielle $\Phi : X \to \mathscr{R}$ telle que $\forall i \in \mathscr{S}$, $\forall x_i, x_i' \in X_i$ et $\forall x_{-i} \in X_{-i}$,*

$$\left|U_i(x_i, x_{-i}) - U_i(x_i', x_{-i}) + \Phi(x_i', x_{-i}) - \Phi(x_i, x_{-i})\right| \leq \xi. \tag{1.3}$$

Pour $\xi = 0$, c'est un jeu potentiel exact [3]. Le ξ capture la différence maximale par paire entre un jeu ξ -potentiel et un jeu potentiel exact avec la même fonction potentielle que dans [4, Définition 2.2].

Comme indiqué précédemment, un jeu potentiel exact a au moins un PNE et les optimiseurs locaux de la fonction potentielle sont les PNE [3]. Dans le lemme suivant, nous fournissons la relation entre les PNE d'un jeu potentiel et un jeu potentiel approché avec le même potentiel.

Lemma 1 *Soient $\mathscr{G} = \left\{\mathscr{S}, \{X_i\}_{i\in\mathscr{S}}, \{U_i\}_{i\in\mathscr{S}}\right\}$ et $\mathscr{G}' = \left\{\mathscr{S}, \{X_i\}_{i\in\mathscr{S}}, \{U_i'\}_{i\in\mathscr{S}}\right\}$ un jeu de potentiel exact et un jeu ξ-potentiel respectivement, partageant une fonction potentielle Φ. Si x^* est une PNE pour $\mathscr{G}$, alors c'est un ξ-NE pour $\mathscr{G}'$.*

Nous présentons maintenant les jeux potentiels bruités qui peuvent modéliser le bruit d'un problème stochastique et ce, afin de les résoudre de façon distribuée. Nous définissons les jeux potentiels bruités ci-dessous.

Definition 4 *[Noisy potential game] Soit l'utilité attendue du joueur i désignée comme $U_i = \mathbb{E}\left[\hat{U}_i\right]$. Le jeu $\hat{\mathscr{G}} \coloneqq \left\{\mathscr{D}, \{X_i\}_{i\in\mathscr{D}}, \left\{\hat{U}_i\right\}_{i\in\mathscr{D}}\right\}$ est un jeu potentiel bruité si le jeu $\mathscr{G} \coloneqq \left\{\mathscr{D}, \{X_i\}_{i\in\mathscr{D}}, \{U_i\}_{i\in\mathscr{D}}\right\}$ est un jeu potentiel avec la fonction potentielle Φ.*

Toutes les conceptions des fonctions de coût qui conduisent à des jeux potentiels conduisent également à des jeux potentiels bruités si la fonction saisit le bruit du problème. En particulier, des jeux potentiels bruités peuvent être obtenus en considérant la fonction d'utilité suivante qui représente la contribution marginale du joueur à la fonction d'utilité globale:

$$\hat{U}_i(x_i, x_{-i}) = \hat{\Phi}\left(x_i, x_{-i}\right) - \hat{\Phi}\left(x_i^0, x_{-i}\right), \tag{1.4}$$

où $\hat{\Phi}$ est une réalisation bruitée d'une fonction objectif Φ et x_i^0 est une action nulle du joueur i. Notez que l'utilité aléatoire $\hat{U}$ peut avoir une grande variance qui conduit le jeu potentiel bruité à avoir une grande déviation par rapport au jeu potentiel exact. Pour réduire la variance de l'utilité, nous pouvons définir un exemple de moyenne de fonction d'utilité:

$$\hat{U}_i^N = \frac{1}{N}\sum_{k=1}^{N}\hat{U}_i. \tag{1.5}$$

Nous verrons dans la prochaine section que le nombre des échantillons N doit être fixé avec soin afin de préserver les propriétés de convergence des jeux potentiels.

Lemma 2 *Pour tout N, le jeu* $\hat{\mathscr{G}}^N := \left\{\mathscr{D}, \{X_i\}_{i\in\mathscr{D}}, \{\hat{U}_i^N\}_{i\in\mathscr{D}}\right\}$ *est un jeu potentiel bruité avec une fonction potentielle* $\Phi(x)$.

1.2.2 Apprentissage distribué dans les jeux potentiels

Rappelons que la fonction de fonction potentielle permet de trouver un optimum global en utilisant des algorithmes d'apprentissage distribués. Dans ce qui suit, nous présentons d'abord l'algorithme Best Response (BR) et l'algorithme d'apprentissage log-linéaire (LLA) pour le cas d'information complète. Ensuite, l'algorithme d'apprentissage log-linéaire binaire (BLLA) pour le cas d'information incomplète est décrit.

Algorithme de la meilleure réponse

L'algorithme de meilleure réponse est un algorithme asynchrone où, à un moment donné, seul un joueur met à jour sa stratégie. Le joueur actuel calcule son coût $U_i(x_i, x_{-i}(t-1))$ pour tous $x_i \in X_i$ compte tenu des stratégies $x_{-i} \in X_{-i}$ d'autres joueurs. Ensuite, le joueur choisit une stratégie $x_i \in X_i$ qui minimise son coût. En d'autres termes, la BS i choisit une stratégie à partir de sa meilleure réponse: B_i:

$$B_i(x_{-i}) = arg\,min_{x_i} U_i(x_i, x_{-i}). \tag{1.6}$$

Notez que l'algorithme BR nécessite des informations complètes, c'est-à-dire l'information de l'ensemble des stratégies du joueur. En outre, l'algorithme BR n'est pas garanti de converger vers le PNE optimal, même dans le jeu potentiel exact, car la fonction potentielle peut avoir plusieurs minima locaux [3]. Pour un jeu ξ-potential, un PNE peut même ne pas exister. Par conséquent, un algorithme BR ne peut pas être utilisé pour un jeu potentiel

approché. Une version généralisée de l'algorithme BR est un algorithme d'apprentissage log-linéaire qui peut échapper à un minimum local et atteindre le minimum global. Cet algorithme est décrit ci-après.

Algorithme d'apprentissage log-linéaire

LLA est un algorithme classique asynchrone qui garantit la convergence au minimum global de la fonction potentielle du jeu potentiel exact [5]. Il a également éte prouvé que LLA convergeait et approximait le NE optimal d'un jeu potentiel approché [4]. Nous le prouvons également dans un contexte plus généralisé utilisant une technique différente. LLA est similaire à BR mais permet des écarts par rapport à la meilleure réponse avec une faible probabilité. Les écarts ou les perturbations sont contrôlés par un paramètre de température τ. Comme la température τ augmente, la perturbation augmente, ce qui entraîne une forte probabilité de choisir une action sous-optimale. Comme la température τ tend vers zéro, la perturbation diminue, ce qui permet à LLA de choisir les meilleures réponses avec une probabilité plus élevée. LLA est résumée dans l'Algorithme 1.

LLA exige également que les joueurs disposent de l'information complète sur les coûts de l'ensembles de leurs actions. Par exemple, compte tenu des stratégies des autres, le joueur doit connaître la valeur de la fonction de coût pour toutes ses stratégies (1.7). Avec cette information, il sélectionne une stratégie à jouer en fonction d'une distribution de probabilité. En général, l'acquisition de cette quantité d'informations n'est pas possible. Pour surmonter cette difficulté dans la prochaine sous-section, nous proposons d'utiliser BLLA.

Algorithm 1 Log-linear Learning Algorithm

1: bf Initialisation: Commencer avec le profil d'action arbitraire x.
2: Définir le paramètre τ.
3: bf While $t \geq 1$ do
4: Sélectionner au hasard un joueur i.
5: Calculer le coût $U_i\left(x_i, x_{-i}(t-1)\right)$ pour tous $x_i \in X_i$.
6: Actionner $x_i(t)$ de X_i avec probabilité $p_i^{x_i}(t)$,

$$p_i^{x_i}(t) = \frac{\exp\left(-\frac{1}{\tau}U_i\left(x_i, x_{-i}(t-1)\right)\right)}{\sum_{x_i' \in X_i}\exp\left(-\frac{1}{\tau}U_i\left(x_i', x_{-i}(t-1)\right)\right)}. \tag{1.7}$$

7: Tous les autres joueurs doivent répéter leurs actions précédentes, c'est-à-dire, $x_{-i}(t) = x_{-i}(t-1)$.

Algorithme d'apprentissage log-linéaire binaire

BLLA est une variante de LLA qui nécessite l'information de seulement deux actions du joueur actuel. Nous disons alors que BLLA est à information partielle ou incomplète. Il s'oppose aux algorithmes à informations complètes où le joueur doit connaître l'effet du choix de toute autre action de son jeu d'actions. BLLA est également un algorithme asynchrone. Dans cet algorithme, le joueur actuel met à jour son action en deux étapes. Dans la première étape, le joueur essaie une action de son jeu d'action pour obtenir son coût. Dans la deuxième étape, le joueur choisit parmi les deux actions (actions présente et d'essai) avec une certaine probabilité telle que résumée dans l'Algorithme 2. BLLA a également un paramètre de température τ qui contrôle la perturbation. Nous disons que l'algorithme *converge* vers un état si la probabilité de l'état est non nulle lorsque la température τ tend vers zéro. Ou, sinon, que l'état est *stochastiquement stable*, voir, par exemple, [5]. Une pratique courante dans la littérature est de trouver empiriquement un paramètre de température approprié τ. Habituellement, la valeur de la température τ dépend des applications spécifiques. Une faible valeur de température peut bien fonctionner dans de nombreuses applications. Nous étudierons l'effet de la température τ en appliquant BLLA à l'équilibrage de charge dans les réseaux sans fil dans le chapitre 1.3. Nous verrons dans le chapitre 1.3.4 un algorithme d'apprentissage pour choisir le meilleur τ.

Algorithm 2 Binary Log-linear Learning Algorithm

1: bf Initialisation: Commencer avec le profil d'action arbitraire x.
2: Définir le paramètre τ.
3: bf While $t \geq 1$ do
4: Choisir de manière aléatoire un joueur i et une action d'essai $\hat{x}_i \in X_i$ avec une probabilité uniforme.
5: Le joueur i joue l'action $x_i(t-1)$ pour calculer le coût U_i $bracx(t-1)$.
6: Le joueur i joue l'action d'essai $\hat{x}_i$ pour calculer son coût $U_i(\hat{x}_i, x_{-i}(t-1))$.
7: Le joueur i sélectionne l'action $x_i(t) \in (x_i(t-1), \hat{x}_i)$ avec une probabilité

$$\left(1 + e^{\Delta_i/\tau}\right)^{-1}, \tag{1.8}$$

où $\Delta_i = U_i(\hat{x}_i, x_{-i}(t-1)) - U_i(x(t-1))$.
8: Tous les autres joueurs répètent leurs actions précédentes, c'est-à-dire, $x_{-i}(t) = x_{-i}(t-1)$.

Soit Φ^* et $\Phi^\dagger$ les premières et secondes valeurs minimum d'une fonction Φ. Soit $|X|$ la cardinalité de l'espace d'action.

Theorem 1 *Pour tout ξ-potential game $\mathscr{G}$ avec potentiel Φ et $\varepsilon > 0$ les états stochastiquement stables de LLA et BLLA correspondent à un ensemble de ξ-NE avec potentiel inférieur à $\Phi^* + 2\xi\,(|X|-1)$ si*

$$\xi < \frac{\varepsilon}{2\,(|X|-1)}. \tag{1.9}$$

Corollary 1 *Pour tout ξ-potential jeu $\mathscr{G}$ avec potentiel Φ les états stochastiquement stables de LLA et BLLA correspondent à un ensemble de PNE dont le potentiel est Φ^* si*

$$\xi < \frac{\Phi^\dagger - \Phi^*}{2(|X|-1)}. \tag{1.10}$$

La preuve de convergence de LLA et BLLA vers un PNE optimal pour un jeu de potentiel exact est donnée dans [5]. En suivant une technique similaire, nous prouvons la convergence de LLA et BLLA vers un minimum global de la fonction potentielle d'un jeu ξ-potentiel $\mathscr{G}$. La preuve de convergence de LLA dans les jeux potentiels approchés est donnée dans [4]. Nous fournissons des preuves à la fois pour LLA et BLLA dans des jeux potentiels approchés en étendant à un cadre plus général en utilisant une technique de preuve différente. Nous utilisons des arbres de résistance des PMC induites. Les expressions de résistance des transitions de LLA et BLLA sont différentes. Nous utilisons nos règles proposées dans la section 1.2.4 pour calculer la résistance. Notre technique n'a pas besoin d'un processus de révision stationnaire. Au lieu de cela, le processus de révision peut dépendre de l'état et de l'historique. Contrairement à [4], notre preuve ne suppose pas que la chaîne de Markov sous-jacente soit réversible, ce qui permet d'étendre leur résultat aux cas où les joueurs sont choisis de manière non stationnaire et dépendante de l'état pour réviser leurs actions. Comme la chaîne de Markov sous-jacente est ergodique, chaque état a une probabilité positive d'être choisi tout au long des itérations de l'algorithme.

1.2.3 Algorithme d'apprentissage distribué dans les jeux potentiels bruités

Nous considérons un jeu potentiel bruité $\hat{\mathscr{G}}^N := \left\{\mathscr{D}, \{X_i\}_{i\in\mathscr{D}}, \left\{\hat{U}_i^N\right\}_{i\in\mathscr{D}}\right\}$ et un jeu potentiel exact correspondant $\mathscr{G} := \left\{\mathscr{D}, \{X_i\}_{i\in\mathscr{D}}, \{U_i\}_{i\in\mathscr{D}}\right\}$ avec fonction potentielle $\Phi(x)$ et $U_i = \mathbb{E}\left[\hat{U}^N\right]$. Nous étudions maintenant BLLA pour l'apprentissage distribué dans des jeux potentiels bruités. L'idée clé de BLLA en présence de bruit consiste à utiliser plusieurs échantillons des fonctions de coûts bruités pour réduire la variance du bruit. En réduisant la variance du bruit dans un jeu potentiel bruité, nous pouvons utiliser les propriétés des jeux

potentiels. Les détails de BLLA en présence de bruit sont décrits dans l'Algorithme 3.

Algorithm 3 BLLA pour les jeux potentiels bruités

1: bf Initialisation: Commencer avec le profil d'action arbitraire x.
2: bf While t $geq1$ do
3: Définir le paramètre $\tau(t)$.
4: Choisir de manière aléatoire un joueur i et une action d'essai $\hat{x}_i \in X_i$ avec une probabilité uniforme.
5: Joueur i joue l'action $x_i(t-1)$ pour N fois et calcule le coût moyen $\hat{U}_i^N(x(t-1))$.
6: Joueur i joue l'action d'essai $\hat{x}_i$ pour N fois pour calculer son coût moyen $\hat{U}_i^N(\hat{x}_i, x_{-i}(t-1))$.
7: À t le joueur i sélectionne l'action $x_i(t) \in (x_i(t-1), \hat{x}_i)$ avec probabilité

$$\left(1+e^{\Delta_i^N/\tau}\right)^{-1}, \tag{1.11}$$

où $\Delta_i^N = \hat{U}_i^N(x(t-1)) - \hat{U}_i^N(\hat{x}_i, x_{-i}(t-1))$.
8: Tous les autres joueurs répètent leurs actions précédentes, c'est-à-dire $x_{-i}(t) = x_{-i}(t-1)$.

Puisque la convergence de BLLA est régie par un paramètre de température τ, nous étudions l'effet de τ et $\tau(t)$ sur la convergence. Notez que dans [6], la convergence de BLLA pour une température fixe est prouvée. Cependant, nous fournissons la preuve du paramètre de température fixe et décroissante de BLLA. En outre, les résultats de la convergence de BLLA pour les deux cas de bruit borné et non borné sont présentés. Comme le paramètre τ passe à zéro, la distribution stationnaire se concentre sur quelques états. Intuitivement, les états dont la probabilité limite est strictement positive quand τ tend vers zéro sont stochastiquement stables comme défini dans la Définition **??**. On sait que pour les jeux potentiels exacts, les états stochastiquement stables de BLLA sont les maximiseurs de la fonction potentielle [5]. Nous étendons ce résultat à des jeux potentiels bruités dans le théorème suivant.

Theorem 2 *Pour tout jeu potentiel bruité $\hat{\mathcal{G}}^N$ avec potentiel Φ les états stochastiquement stables de BLLA sont les maximiseurs globaux du potentiel Φ si l'une des situations suivantes est valide.*

1. *Le bruit d'estimation est borné dans un intervalle de taille ℓ et le nombre d'échantillons d'estimation utilisé est*

$$N \geq \left(\log\left(\frac{4}{\xi}\right)+\frac{2}{\tau}\right)\frac{\ell^2}{2(1-\xi)^2\tau^2}, \tag{1.12}$$

où $0 < xi < 1$.

2. *Le bruit d'estimation est non borné avec moyenne et variance finie. Soit* $M(\theta)$ *la fonction génératrice des moments du bruit, supposée finie. Soit* $\theta^* = arg\,max_\theta\,\theta\,(1-\xi)\,\tau - \log(M(\theta))$. *Le nombre d'échantillons utilisé est*

$$N \geq \frac{\log\left(\frac{4}{\xi}\right) + \frac{2}{\tau}}{\log\left(\frac{e^{\theta^*(1-\xi)\tau}}{M(\theta^*)}\right)}. \tag{1.13}$$

Le corollaire suivant suit immédiatement dans le cas où le bruit est supposé avoir une distribution gaussienne.

Corollary 2 *Supposons que le bruit ait une distribution de probabilité normale. Les états stochastiquement stables de BLLA sont les maximiseurs globaux de la fonction potentielle si*

$$N \geq \frac{2\log\left(\frac{4}{\xi}\right) + \frac{4}{\tau}}{\tau^2\,(1-\xi)^2}. \tag{1.14}$$

Un petit N est souhaité pour les implémentations pratiques. Nous pouvons donc choisir le plus bas N qui satisfait le Théorème 2. Dans le Théorème 2, nous avons une convergence en probabilité pour le paramètre fixe τ. Dans le Théorème 3, nous considérons le cas du paramètre décroissant τ pour lequel nous obtenons une convergence presque sûre à l'état optimal comme dans le recuit simulé avec le même schéma de recuit [7].

Theorem 3 *Considérons BLLA dans un jeu de potentiel bruité avec un paramètre décroissant* $\tau(t) = 1/\log(1+t)$, *et le nombre d'échantillons* $N(\tau)$ *donné par Theorem 2. Alors BLLA converge avec une probabilité* 1 *vers un maximiseur global de la fonction potentielle.*

La dynamique des algorithmes d'apprentissage proposés est analysée en utilisant les arbres d'une chaîne de Markov perturbée. Maintenant nous présentons les règles proposées pour calculer la résistance des arbres. Ces règles seront utilisées pour les preuves de convergence des algorithmes proposés dans les sections suivantes.

1.2.4 Règles de calcul de la résistance

Definition 5 (Résistance d'une fonction positive) *La résistance d'une fonction strictement positive* $f(\tau)$ *est* $Res(f)$ *s'il existe une fonction strictement positive* $g(\tau)$ *telle que*

$g \in o\left(e^{k/\tau}\right)$ *et* $g \in \omega\left(e^{-k/\tau}\right)$ *pour tout* $k > 0$*; et*

$$\lim_{\tau \to 0} \frac{f(\tau)}{g(\tau) e^{-\frac{Res(f)}{\tau}}} = 1. \tag{1.15}$$

La proposition suivante fournit des règles importantes pour calculer $\mathrm{Res}(f)$.

Proposition 4 *Soient* f, f_1 *et* f_2 *des fonctions strictement positives. Soient* $Res(f_1)$ *et* $Res(f_2)$ *supposées bien définies. Soit kappa une constante positive.*

I $f_1(\tau)$ *est sous-exponentielle si et seulement si* $Res(f_1) = 0$*. En particulier* $Res(\kappa) = 0$*.*

II $Res(e^{-\kappa/\tau}) = \kappa$*.*

III $Res(f_1 + f_2) = \min\{Res(f_1), Res(f_2)\}$*.*

IV Si $Res(f_1) < Res(f_2)$ *donc* $Res(f_1 - f_2) = Res(f_1)$*.*

V $Res(f_1 f_2) = Res(f_1) + Res(f_2)$*.*

VI $Res(\frac{1}{f}) = -Res(f)$*.*

VII Si $\forall \tau, f_1(\tau) \leq f_2(\tau)$ *donc* $Res(f_2) \leq Res(f_1)$*.*

VIII Si $\forall \tau, f_1(\tau) \leq f(\tau) \leq f_2(\tau)$ *et si* $Res(f_1) = Res(f_2)$ *alors* $Res(f)$ *existe et* $Res(f) = Res(f_1)$*.*

Remark Dans la règle IV, si $\mathrm{Res}(f_1) = \mathrm{Res}(f_2)$, nous ne pouvons pas calculer $\mathrm{Res}(f_1 - f_2)$ car en général la différence des fonctions sous-exponentielles peut ne pas être une fonction sous-exponentielle. Par exemple, choisissez $f_1(\tau) = 1 + e^{-k/\tau}$ et $f_2(\tau) = 1$ avec $k > 0$ puis $\mathrm{Res}(f_1) = \mathrm{Res}(f_2) = 0$ mais $\mathrm{Res}(f_1 - f_2) = k$.

Remark Pour la règle VIII, en général si $f_1(\tau) \leq f(\tau) \leq f_2(\tau)$ et $\mathrm{Res}(f_1) \neq \mathrm{Res}(f_2)$ alors $\mathrm{Res}(f)$ peut ne pas exister. Par exemple, pour $f(\tau) = \lambda(\tau) f_1 + (1 - \lambda(\tau)) f_2$, $\lambda(\tau) = \frac{1}{2}\left(\cos\left(\frac{1}{\tau}\right) + 1\right)$ $\mathrm{Res}(f)$ n'existe pas.

1.3 Équilibrage de charge dans les réseaux sans fil hétérogènes

Dans cette section, nous présentons une nouvelle approche pour l'équilibrage de charge distribué dans les réseaux hétérogènes qui utilisent le paramètre d'extension de couverture (CRE) pour l'association d'utilisateurs et le partage en temps (ABS) pour la gestion des interférences. Nous adaptons les algorithmes d'apprentissage dans les jeux potentiels approché décrits dans la section précédente pour résoudre le problème d'équilibrage de charge.

En raison de la demande toujours croissante pour améliorer la qualité du service en termes de débits et de couverture, les réseaux cellulaires classiques deviennent hétérogènes. Les réseaux hétérogènes se composent de stations de base macro (BS) et de petites BS qui transmettent avec une puissance élevée et faible, respectivement. La règle conventionnelle d'association d'utilisateurs est telle que les utilisateurs choisissent une BS qui fournit la plus haute puissance reçue. Cela peut cependant entraîner un déséquilibre entre les charges des BS car les BS macro transmettent à une puissance supérieure et associent ainsi plus d'utilisateurs. Cela crée une situation de surcharge aux BS macro et, en même temps, des ressources sous-utilisées aux petites BS. Par conséquent, un problème naturel qui se pose est la façon d'associer les utilisateurs aux BS, de sorte que les ressources du réseau soient utilisées efficacement et que la charge soit partagée entre les BS.

L'équilibrage de charge a été largement étudié dans la littérature en utilisant diverses approches. ăUne vue d'ensemble peut être trouvée dans [8, 9]. Celles-ci peuvent être classées globalement en approches centralisées centrées sur le réseau et approches décentralisées centrées sur les utilisateurs, voir par exemple [10–14].

Nous nous concentrons sur une approche alternative distribuée centrée sur le réseau, dans laquelle les BS prennent des décisions et les utilisateurs suivent une règle d'association prédéfinie appelée Cell Range Extension (CRE). Selon la technique CRE, les utilisateurs s'associent à une BS qui fournit la puissance reçue biaisée maximale. Un biais CRE est diffusée par chaque BS et est généralement plus élevé pour les petites BS que pour les BS macro. Il en résulte une augmentation de la couverture des petites cellules et donc du nombre d'utilisateurs qui y sont associés. La technique CRE a l'inconvénient d'une probabilité de dépassement du rapport signal à interférence plus bruit (SINR) croissante au bord de la cellule [15], elle est donc souvent déployée conjointement avec des sous-trames presque vierges (ABS) à la macro BS [16]. Lorsque ces sous-trames, qui représentent une proportion fixe de la trame radio, sont transmises, les BS macro réduisent considérablement

leur puissance d'émission, de sorte que les utilisateurs de bord de cellule des petites BS peuvent subir moins d'interférences lorsqu'ils sont ordonnancés pendant ces périodes.

Modèle de réseau

Nous considérons la voie descendante [1] d'un réseau cellulaire (généralement un réseau LTE-Advanced) constitué d'un ensemble $\mathscr{B}_e$ de macro BS (généralement eNodes-B) et un ensemble $\mathscr{B}_s$ de petites BS dans une région bi-dimensionnelle $\mathscr{L}$. L'ensemble de toutes les stations est désigné par $\mathscr{S} \triangleq \mathscr{B}_e \cup \mathscr{B}_s$. Les puissances de transmission des macro et petites BS sont désignées par P_{macro} et P_{small}, respectivement. Il existe des sous-trames spéciales appelées ABS, pendant lesquelles une macro BS transmet avec une puissance réduite P_{ABS}. La proportion de sous-trames ABS est notée $\theta_i \in [0;1]$ pour BS i. Soit $\bar{\boldsymbol{\theta}} = \left[\theta_1, \theta_2, \ldots, \theta_{|\mathscr{S}|}\right]$ le vecteur de ratios ABS. Nous supposons que les petites BS ne mettent pas en oeuvre la technique ABS, c'est-à-dire $\theta_i = 0$ pour $i \in \mathscr{B}_s$. Chaque petite BS i maintient un paramètre $c_i \in [1; c_{\max}]$ appelé le biais CRE. Le vecteur de biais CRE est désigné par $\bar{\boldsymbol{c}} = [c_1, c_2, \ldots, c_{|\mathscr{S}|}$. Les biais CRE pour les BS macro sont fixés à l'unité, c'est-à-dire $c_k = 1, \forall k \in \mathscr{B}_e$. Cela ne conduit à aucun biais dans la puissance reçue d'une macro BS.

La puissance reçue à l'emplacement x de la BS i est $P_i g_i(x)$, où P_i est la puissance d'émission et $g_i(x)$ est le gain de canal, qui prend en compte l'affaiblissement de parcours et l'effet de masque.

Règle d'association des utilisateurs fondée sur le CRE

Une règle d'association d'utilisateur basée sur le paramètre CRE et la puissance d'émission maximale est couramment utilisée dans les réseaux hétérogènes [8, 15, 16, 18–25]. Selon cette règle, un utilisateur situé à x est associé à la BS i qui fournit la puissance reçue biaisée la plus élevée. L'ensemble des emplacements $\mathscr{D}_i(\bar{\boldsymbol{c}})$ associé à la BS i est défini comme suit:

$$\mathscr{D}_i(\bar{\boldsymbol{c}}) = \left\{x | \forall j \in \mathscr{S}, P_i g_i(x) c_i \geq P_j g_j(x) c_j\right\}, \tag{1.16}$$

où $P_i = P_{macro}$ si $i \in \mathscr{B}_e$ et P_{small} autrement.

[1] La voie descendante est généralement considérée comme le lien dominant en termes de trafic. Cependant, une association optimale d'utilisateurs sur le lien descendant peut ne pas être optimale pour le lien montant [17].

Formulation du problème et fonction objectif

Suivant [10], nous avons l'intention de minimiser une fonction d'$\alpha-$équité $\phi_\alpha\left(\bar{c},\bar{\theta}\right)$ sur un ensemble faisable $\mathscr{F}$, qui sont définis comme suit:

$$\phi_\alpha(\bar{c},\bar{\theta})=\begin{cases}\sum_{i\in\mathscr{S}}\frac{(1-\rho_i(\bar{c},\bar{\theta}))^{1-\alpha}}{\alpha-1}, & \alpha\geq 0,\alpha\neq 1,\\ -\sum_{i\in\mathscr{S}}\log\left(1-\rho_i(\bar{c},\bar{\theta})\right), & \alpha=1,\end{cases} \tag{1.17}$$

$$\mathscr{F}=\left\{\left\{\bar{c},\bar{\theta}\right\}|\forall i\in\mathscr{S},\rho_i(\bar{c},\bar{\theta})<1,O_i(\bar{c},\bar{\theta})<\bar{O}_i,\right\}, \tag{1.18}$$

où $\bar{O}_i$ est la probabilité de dépassement maximale pour la BS i. La fonction $\phi_\alpha(\bar{c},\bar{\theta})$ est en général non convexe et même si elle était convexe l'ensemble $\mathscr{F}$ n'est pas convexe car $\bar{c}$ prend des valeurs discrètes. La function $\phi_\alpha(\bar{c},\bar{\theta})$ capture différents aspects de l'équité et des performances pour le réseau en fonction du choix de $alpha$.

$(\alpha=0)$ **Politique de charge de somme minimale:** Minimiser $\phi_0(\bar{c},\bar{\theta})$ minimise la somme des charges des BS en général. Dans le cas particulier, où $\bar{\theta}=0$, il en résulte une politique optimale en termes de débits.

$(\alpha=1)$ **Politique équitable proportionnelle:** Minimiser $\phi_1(\bar{c},\bar{\theta})$ est équivalent à l'équité proportionnelle entre les BS [1]. $(\alpha=2)$ **Politique de délai minimale:** On peut montrer que la minimisation de $\phi_2\left(\bar{c},\bar{\theta}\right)$ équivaut à minimiser le débit moyen du réseau. Le débit moyen d'une file d'attente stable M / G / 1 / PS est le produit du taux d'arrivée et du délai moyen. Dans notre modèle de système, le taux d'arrivée est indépendant du biais de la CRE et du ratio ABS. Par conséquent, la réduction de $\phi_2\left(\bar{c},\bar{\theta}\right)$ équivaut à minimiser le délai moyen du réseau. Pour une discussion plus détaillée, se référer à [10]. ă $(\alpha\to\infty)$ **Politique Minmax:** Comme $\alpha\to\infty$ la minimisation de $\phi_\alpha(\bar{c},\bar{\theta})$ tend vers le vecteur de charge min-max . C'est un résultat standard avec une fonction objectif convexe [1,10,26]. Nous étendons ce résultat à notre fonction objectif non convexe dans le Théorème 5.

Definition 6 *[Vecteur de charge Min-max [26]] Supposons que tous les vecteurs de $\mathscr{F}$ soient triés en ordre croissant. Un vecteur $\rho\in\mathscr{F}$ est min-max si ρ est lexicographiquement pas plus grand que n'importe quel vecteur dans $\mathscr{F}$. Le vecteur ρ est* lexicographiquement inférieur *à y, noté $\rho\prec y$, si la première composante non nul de ρ_y est négative. Nous disons que ρ n'est pas supérieur à y, noté $\rho\preceq y$, si $\rho\prec y$ ou $\rho=y$.*

Soit $r_i(\bar{c})=1-\rho_i(\bar{c},\bar{\theta}),\forall i\in\mathscr{S}$. Soit $\mathscr{X}=\left\{r\in\mathscr{R}^{|\mathscr{S}|}|\exists\bar{c}:\rho(\bar{c})\in\mathscr{F},r(\bar{c})=r\right\}$. Le vecteur de charge ρ^* est un min-max si et seulement si r^* est le vecteur max-min.

Theorem 5 *Soit* $r_\alpha \in \underset{r\in\mathscr{X}}{argmax} \sum_{i\in\mathscr{S}} \frac{r_i^{1-\alpha}}{1-\alpha}$. *Alors tout vecteur d'accumulation de la trajectoire* $\{r_\alpha\}_{\alpha>1}$ *est max-min dans* $\mathscr{X}$.

Nous modélisons maintenant les interactions des BS pour l'équilibrage de charge en tant que jeu potentiel approché.

Un *jeu d'association des utilisateurs* est un jeu fini avec les BS en tant que joueurs et leurs stratégies sont des valeurs admissibles du biais CRE et du ratio ABS. Nous concevons les fonctions de coûts des BS pour modéliser le jeu d'association d'utilisateurs en tant que jeu potentiel approché. Pour cela, nous imposons que la fonction objectif (1.25) soit transformée en une fonction potentielle. Nous parlons de façon interchangeable d'un jeu potentiel approché ou d'un jeu ξ-potentiel, où le paramètre ξ représente un compromis entre la qualité de la solution et la nature distribuée de l'algorithme. Le problème est de concevoir les fonctions de coûts des BS pour obtenir un jeu ξ -potentiel que nous abordons ci-dessous.

Nous considérons une structure de coûts simple pour la BS i, qui ne prend en compte que les effets de ses voisins. Une approche similaire est utilisée pour obtenir un jeu de potentiel exact dans [27, 28]. Les fonctions de coût des BS individuelles sont définies comme suit:

$$U_i^{\varpi}(a_i, a_{-i}) = \sum_{j\in N_i^{\varpi}} \frac{(1-\rho_j(a_i,a_{-i}))^{1-\alpha}}{\alpha-1}, \tag{1.19}$$

où $\rho_j(a_i, a_{-i})$ est la charge de la BS j donnée, N_i^{ϖ} est le voisinage de la BS i, et ϖ est un paramètre pour contrôler la taille du voisinage. L'effet de ϖ sur le voisinage N_i^{ϖ} est comme suit:

$$\begin{aligned} \varpi = 0 &\implies N_i^0 = \mathscr{S}, \\ \varpi = 1 &\implies N_i^1 = BS\ i, \\ 0 < \varpi < 1 &\implies N_i^{\varpi} \subset \mathscr{S}. \end{aligned}$$

La taille du voisinage N_i^{ϖ} dépend également de la décorrélation des masques.

Dans ce qui suit, nous définissons formellement un jeu d'association d'utilisateurs, puis montrons qu'il s'agit d'un jeu ξ-potentiel. Notez qu'un jeu de potentiel exact est également un jeu ξ-potentiel avec $\xi = 0$.

Definition 7 *[Jeu d'association des utilisateurs] C'est un jeu* $\Gamma^{\varpi} := \left\{\mathscr{S}, \{X_i\}_{i\in\mathscr{S}}, \left\{U_i^{\varpi}\right\}_{i\in\mathscr{S}}\right\}$, *où* $\mathscr{S}$ *est un ensemble de BS,* X_i *est un ensemble de stratégies de BS i et* U_i^{ϖ} *est une fonction*

de coût donnée dans (1.19). Le jeu de stratégie X_i est un ensemble discret des valeurs de biais CRE pour les petites BS $i \in \mathscr{B}_s$ et X_i est un ensemble discret de ratios ABS pour la macro BS $i \in \mathscr{B}_e$.

Proposition 6 *Le jeu d'association d'utilisateurs Γ^{ϖ} avec la fonction de coût (1.19) est un jeu ξ-potentiel avec la fonction potentielle (1.25), où ξ est une fonction de ϖ:*

$$\xi = \max_{a_i, a_i' \in X_i, a_{-i} \in X_{-i}, i \in \mathscr{S}} \left| \sum_{j \in N_i^0 \setminus N_i^{\varpi}} \frac{\left(1-\rho_j(a_i, a_{-i})\right)^{1-\alpha}}{\alpha - 1} - \sum_{j \in N_i^0 \setminus N_i^{\varpi}} \frac{\left(1-\rho_j(a_i', a_{-i})\right)^{1-\alpha}}{\alpha - 1} \right|. \tag{1.20}$$

Corollary 3 *Le jeu Γ^0 est un jeu de potentiel exact.*

1.3.1 Les algorithmes d'apprentissage distribués

Dans cette section, nous adaptons le BLLA distribué développé pour le jeu ξ-potentiel Γ^{ϖ} pour trouver l'optimum de la fonction potentielle ϕ_α. Nous avons montré dans le Théorème 1 que, dans certaines conditions, ces deux algorithmes convergent vers un PNE optimal pour le jeu ξ-potentiel. Nous obtenons des conditions suffisantes en fonction du paramètre système afin de respecter les conditions du théorème 1.

Soient ϕ_α^* et $\phi_\alpha^\dagger$ les première et seconde valeurs minimum du potentiel. Le théorème suivant donne une limite supérieure sur ϖ qui garantit que LLA et BLLA convergent vers un PNE optimal.

Theorem 7 *La contrainte dans le Théorème 1 est satisfaite si*

$$\varpi \leq \varepsilon Q \left(1-\rho_{\max}\right)^{\alpha}, \tag{1.21}$$

où $\rho_{\max}$ est la charge maximale possible d'une BS, $Q = \frac{\max_{x,\bar{\theta},j\in\mathscr{S}} \frac{1}{\nu_j(x,\bar{\theta})}}{4|\mathscr{S}||X|\lambda_m \max_x\left\{\frac{1}{\mu(x)}\right\}}$, et λ_m est une limite supérieure pour le taux d'arrivée dans une cellule.

Corollary 4 *La contrainte du corollaire 1 est satisfaite si*

$$\varpi \leq Q\left(\phi_\alpha^\dagger - \phi_\alpha^*\right)\left(1-\rho_{\max}\right)^{\alpha}. \tag{1.22}$$

Table 1.1: Paramètres de simulations.

Paramètre	Variable	Valeur
Nombre de BS	N_s	8
Macro BS pendant NS	P_{macro}	46 dBm
Macro BS pendant ABS	P_{ABS}	0 dBm
Petite BS	P_{small}	24 dBm
Taille moyenne du fichier	$\frac{1}{\mu}$	0.5 Mbytes
Densité moyenne de la charge de trafic	$\frac{\lambda}{\mu}$	64 bits$/s/m^2$
Bande passante du système	W	20 MHz
Fréquence porteuse	f_c	2.6 GHz
Puissance de bruit	N	-174+10log(W) dBm
Minimum SINR	γ_{min}	-10 dB
Exposant d'affaiblissement	η	3.5
Distance de référence	d_0	10 m
Ensemble de biais CRE	c_i	$\{1, 1.1, 1.2, \ldots, 16\}$
Ratio ABS	θ_i	$\{0, 0.01, 0.02, \ldots, 1\}$

1.3.2 Résultats de simulations

Dans cette section, nous montrons les résultats de simulation de l'équilibrage de charge dans des réseaux hétérogènes en utilisant les algorithmes proposés. Nous considérons d'abord l'effet de masque et utilisons les biais CRE pour l'équilibrage de charge. Ensuite, dans le même scénario, nous utilisons à la fois le biais de CRE et l'ABS pour montrer clairement l'effet de l'ABS sur l'équilibrage de charge.

Nous considérons les paramètres standards adoptés dans 3GPP [29]. Ces paramètres sont répertoriés dans la table 1.1. Nous considérons 8 BS dans une région bi-dimensionnelle $\mathscr{L}$. BS 1 est une macro BS qui transmet avec P_{macro} et le reste sont de petites BS qui transmettent avec P_{small}. Le trafic utilisateur varie avec l'emplacement avec une densité de trafic moyenne de 64 bits $/s/m^2$. Il y a deux points chauds où le trafic est 5 fois plus élevé que le trafic moyen, qui peut être visualisé dans la Fig. 1.1. Nous utilisons la formule classique de Shannon pour calculer la capacité du canal $\tilde{v}_i^f(x, \bar{\boldsymbol{\theta}}) = W \log_2\left(1 + \gamma_i^f(x, \bar{\boldsymbol{\theta}})\right)$.

Nous utilisons le jeu potentiel approché Γ^{ϖ} pour ce scénario car il permet d'utiliser le voisinage réduit N_i^{ϖ}. Nous considérons un écart type de l'effet de masque de $\sigma_{sh} = 8$ dB et une distance de décorrélation de $D_c = 20$ m. La corrélation croisée entre les composants de l'effet de masque à un emplacement est considérée comme étant de 0.5. Nous nous concentrons d'abord sur l'optimisation du CRE et supposons dans cette section que la macro BS ne met pas en oeuvre ABS ($\theta_1 = 0$). Nous considérons une région carrée $\mathscr{L}$ du

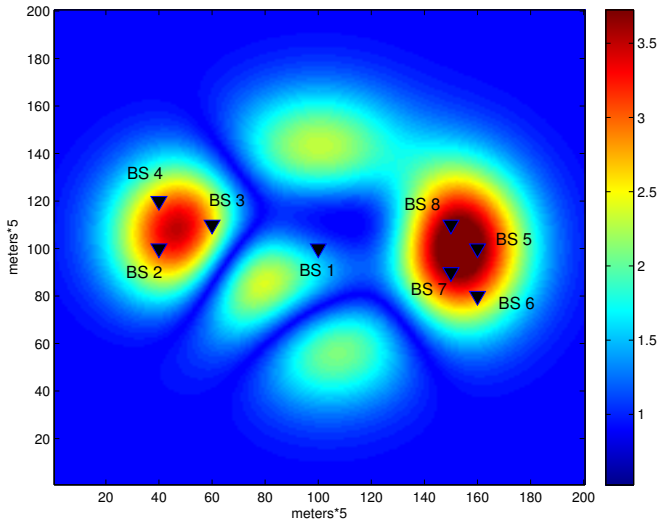


Figure 1.1: Cette figure montre des variations de trafic normalisées dans une région d'un kilomètre carré. Il y a 8 BS, où BS 1 est macro et les autres sont des BS de petite taille situées autour de deux points chauds de trafic.

côté 1000 m.

La convergence de LLA et BLLA en utilisant les valeurs de paramètre $\tau = 10^{-3}$, $\varpi = 10^{-22}$ est illustrée dans les Figures 1.2 pour $\alpha = 50$. Nous observons que dans tous ces modèles, les algorithmes proposés convergent en quelques dizaines et parfois quelques centaines d'itérations. Nous voyons également que LLA converge plus rapidement que BLLA. Ceci est évidemment dû à l'ensemble des informations disponibles pour LLA. Notez cependant que BLLA ne perd pas tellement en termes de vitesse de convergence; c'est une conclusion intéressante pour une mise en oeuvre pratique. Le vecteur de charge optimal obtenu est donné dans le tableau 1.2.

1.3.3 Equilibrage de charge avec CRE et ABS et effet de masque

Dans cette section, nous autorisons la macro BS à mettre en oeuvre l'ABS et étudions l'effet de cette technique sur l'équilibrage de charge. Par souci de simplicité, nous établissons $\tau = 10^{-3}$, $\varpi = 10^{-22}$ et utilisons LLA. Nous considérons également une région carrée $\mathscr{L}$ de côté 2000 m dans les simulations. Trois cas peuvent être comparés pour évaluer l'intérêt de l'utilisation de l'ABS:

1. *Aucune contrainte de dépassement sans ABS:* Dans ce cas, la macro BS n'implémente pas ABS ($\theta_1 = 0$) et nous n'imposons pas une contrainte de dépassement. Cela sert

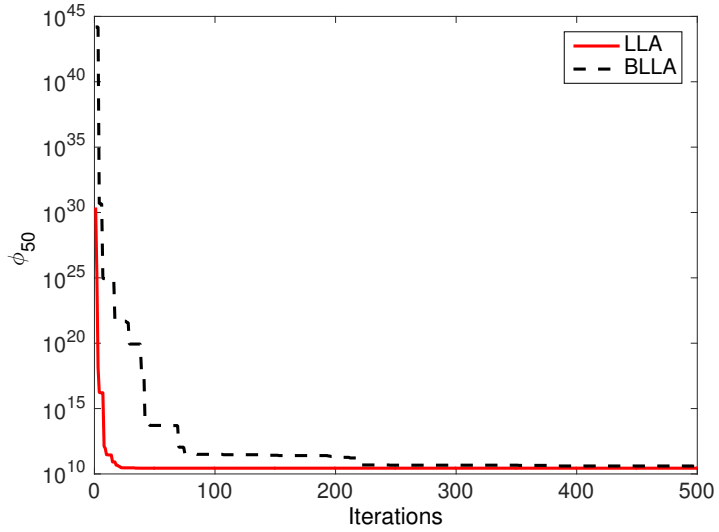


Figure 1.2: [($\alpha = 50$) Polique Min-max]: Cette figure compare la convergence de LLA et BLLA en utilisant $\tau = 0,001$ et $\varpi = 10^{-22}$. Dans ce cas, les valeurs de la fonction objectif sont exponentiellement importantes. LLA et BLLA convergent vers le même minimum. BLLA prend plus d'itérations que LLA.

Table 1.2: Comparaison de CRE optimaux, charges optimales de BS pour différentes α ($\tau = 10^{-3}$, $\varpi = 10^{-22}$).

	$\alpha = 0$		$\alpha = 2$		$\alpha \to \infty$	
BS i	c_i^*	$\rho_i^*\%$	c_i^*	$\rho_i^*\%$	c_i^*	$\rho_i^*\%$
1	1	92	1	61	1	45
2	1	7	3	20	8	42
3	1	4	3	9	9	23
4	1	9	3	18	8	37
5	1	11	3	21	7	37
6	1	8	3	20	7	43
7	1	5	3	11	8	30
8	1	7	3	19	6	37

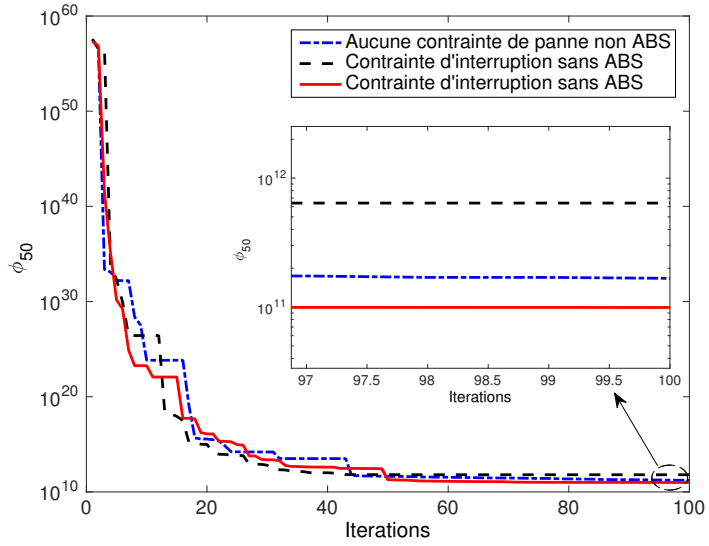


Figure 1.3: [Effet de la contrainte de dépassement et de l'ABS sur l'équilibrage de charge pour $\alpha = 50$]: Il existe une énorme différence dans le coût global optimal dans ce cas, car le biais optimal de toutes les BS est plus élevé, ce qui entraîne des dépassements plus élevés. Nous pouvons voir que le coût global est beaucoup plus petit avec ABS par rapport à celui sans ABS.

de référence pour comparer avec les deux autres cas.

2. *Contrainte de dépassement sans ABS:* Nous présentons ici une contrainte de dépassement ($\bar{O}_i = 2\%$ pour tout i) mais n'autorisons toujours pas ABS ($\theta_1 = 0$). La panne est prise en compte en fixant un coût infini pour les actions qui violent la contrainte $\bar{O}_i$.

3. *Contrainte de dépassement avec ABS:* Nous imposons une contrainte de dépassement et autorisons l'ABS à la macro BS.

1.3.4 Paramètre d'apprentissage τ de LLA

Le paramètre τ des algorithmes d'apprentissage doit être choisi avec soin pour obtenir les performances souhaitées. Dans cette section, nous proposons un nouvel algorithme d'apprentissage de recuit distribué (ALA) donné dans Algorithm 4 pour apprendre automatiquement le paramètre de LLA. ALA est développé en adaptant l'algorithme de rejet successif (SRA) [2]. Dans ALA, un horizon de temps est considéré et un ensemble de valeurs de τ de large gamme est considéré comme exécuté le LLA. Dans l'évolution de

LLA, ALA vérifie la performance de différents τ pour rejeter successivement ceux avec la pire performance. À la fin de l'horizon, ALA sélectionne le τ performant. Le processus de rejet successif de l'ensemble des mauvais résultats τ dans un calendrier de recuit pour LLA. Nous verrons que ce calendrier de recuit est rapide, fonctionne mieux que d'autres horaires de recuit et garantit une convergence asymptotique. Nous appliquons ALA pour le problème d'équilibrage de charge en utilisant uniquement l'association de polarisation CRE sous le modèle de canal pathloss. Nous ignorons la technique ABS afin de simplifier le modèle du système.

Algorithm 4 Algorithme D'apprentissage des Paramètres

1: **Initialisation:** Laissez T être un horizon temporel et M soit le nombre de τ paramètres. $A_1 = \{\tau_1, \ldots, \tau_M\}$ and $\Upsilon = \frac{1}{2} + \sum_{i=2}^{M} \frac{1}{i}$.
2: **for** Phase $k = 1, 2, \ldots, M-1$ **do**
3:
$$n_k = \left\lceil \frac{1}{\Upsilon} \frac{T-M}{M+1-k} \right\rceil \tag{1.23}$$
et $n_0 = 0$.
4: **for** Pour tout $i \in A_k$ **do**
5: Joyez LLA avec τ_i pour $n_k - n_{k-1}$ iterations.
6: **end for**
7: Calculer
$$A_{k+1} = A_k \backslash arg\,max_{i \in A_k} \hat{X}_{i,n_k}, \tag{1.24}$$
où $\hat{X}_{i,n_k} = \frac{1}{n_k} \sum_{t=1}^{n_k} X_{i,t}$, where $X_{i,t}$ est le coût obtenu en utilisant (1.19) avec τ_i à l'instant t.
8: **end for**

La performance d'ALA avec $\tau = \left[10^{12}, 10^{10}, 10^{8}, 10^{2}\right]$ est affichée dans Fig. 1.4 pour $\alpha = 50$. Dans les premières itérations, beaucoup d'oscillations sont observées. Ceci est dû aux valeurs élevées de τ utilisées par ALA pour explorer tous les paramètres τ donnés. À mesure que l'algorithme progresse, les valeurs supérieures de τ sont rejetées et l'algorithme progresse vers la stabilisation. À la fin, l'ALA exécute LLA avec le $\tau = 10^8$ survivant et il converge. Nous voyons que le plus petit $\tau = 10^2$ n'est pas toujours le meilleur paramètre puisque LLA peut être coincé dans le minimum local avec τ.

1.4 Affectation de canal dans les réseaux sans fil D2D

Dans ce section, nous présentons une application de l'algorithme d'apprentissage logarithmique logarithmique développé (BLLA) pour les jeux potentiels bruyants. Nous appliquons

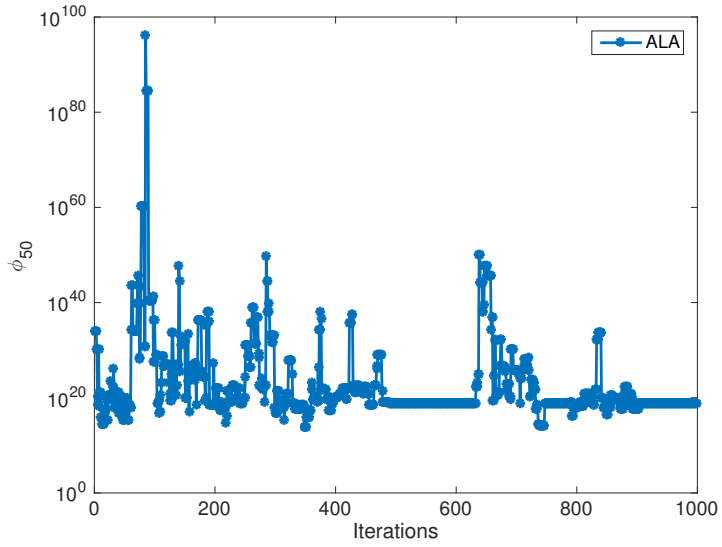


Figure 1.4: Evolution de LLA avec ALA avec $\boldsymbol{\tau} = \left[10^{12}, 10^{10}, 10^{8}, 10^{2}\right]$. Dans les premières itérations il y a beaucoup de fluctuations puisque ALA explore toutes les valeurs de τ. Comme ALA rejette les valeurs de τ les moins performantes, les fluctuations diminuent. Le paramètre le plus performant $\tau = 10^{8}$ est choisi par ALA à la fin de l'horizon temporel.

BLLA pour Channel Assignment Problem (CAP) dans les réseaux sans fil Device-to-Device (D2D).

La demande croissante pour les taux de données plus élevés des utilisateurs mobiles et la rareté du spectre de fréquences sans fil rend les ressources de spectre d'utilisation efficaces de plus en plus critiques. Les réseaux Device-to-Device (D2D) augmentent l'utilisation des ressources du spectre en fournissant une réutilisation du spectre spatial [**?**, 30–40]. En outre, les réseaux D2D fournissent le déchargement du réseau cellulaire, la réduction des coûts de communication et les pouvoirs de calcul entre les dispositifs [41], fournissent un réseau secondaire ou de secours en cas de dommages au réseau cellulaire pendant une catastrophe naturelle, Fournir de nombreuses applications telles que le partage de flux vidéo événementiel et les services contextuels [41]. Un réseau D2D se compose de stations de base (BS), d'équipements utilisateurs cellulaires (UE) et d'utilisateurs D2D. Les UE peuvent être cellulaires ou D2D selon la sélection de mode. Les UE mobiles communiquent sur les bandes de licences avec une BS de la cellule de couverture tandis que les UE D2D réutilisent les bandes autorisées sans ou avec l'aide limitée des BS. Cependant, la réutilisation des chaînes crée des interférences co-canal qui dégradent la qualité du service (QoS) des UE cellulaires. Par conséquent, un problème crucial dans les réseaux D2D sous-jacents est d'attribuer des canaux aux UE afin que l'interférence totale par canal soit

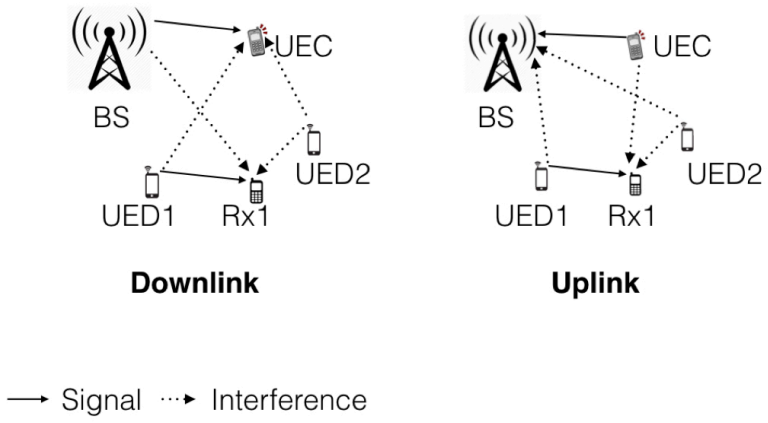


Figure 1.5: [Modèle de réseau cellulaire D2D]: cette figure montre les liaisons de signal et d'interférence dans la liaison descendante et le liaison montante. Dans la liaison descendante, les utilisateurs D2D UED1 et UED2 provoquent des interférences co-canal à UEC utilisateur cellulaire. De plus, BS et UED2 provoquent des interférences co-canal au RX1. Alors que, en liaison montante, UED1 et UED2 provoquent des interférences à BS; et UEC et UED2 provoquent des interférences à RX1.

faible.

1.4.1 Modèle de Réseau Cellulaire D2D

La figure 1.5 montre les liens descendants (DL) et les uplinks (UL) d'un modèle de réseau D2D. Nous considérons une station de base unique (BS) et deux types d'UE: (i) UE cellulaires (UEC) qui communiquent avec les BS et (ii) D2D UE (UED) qui communiquent avec d'autres UED. Tous les UE sont situés au hasard dans la région. L'ensemble des UE est désigné par $\mathscr{D}$. Nous considérons un ensemble de bandes de chaînes de fréquences orthogonales $\mathscr{F}$. Les UEC reçoivent des canaux différents par BS, alors que les UED réutilisent ces canaux. Un UE transmet sur un seul canal. Les UE qui transmettent sur la même chaîne $c \in \mathscr{F}$ provoquent des interférences entre eux. Dans DL, BS transmet un signal à UEC tout en provoquant des interférences sur les récepteurs d'UED qui utilisent le même canal que dans Fig. 1.5. Un UED transmet un signal à son récepteur (Rx) tout en provoquant des interférences aux UEC et aux récepteurs d'autres UED qui se trouvent sur le même canal. En UL, UEC transmet un signal à la BS tout en provoquant des interférences aux UED, voir Fig. 1.5. En outre, les UED provoquent des interférences avec les BS et

d'autres UED dans UL. La quantité d'interférence causée dépend du gain de canal entre les périphériques. Le modèle de canal est décrit ci-dessous.

Nous considérons un modèle de canal qui capte l'effet de pathloss, d'ombre et de décoloration à petite échelle.

1.4.2 Formulation du Problème

Notre objectif est de maximiser le débit de données de somme attendu du réseau D2D en assignant de manière optimale des canaux aux UE. Soit $\bar{c} = (c_i, c_{-i})$ désigne un vecteur d'affectation de canal où UE i reçoit le canal $c_i \in \mathscr{F}$ et les UE autres que l'UE i reçoit le vecteur de canal $c_{-i} \in \mathscr{F}^{|\mathscr{D}|-1}$. Le taux de données estimé d'un UE dépend du vecteur $\bar{c}$ et est désigné par $\hat{\nu}_j(\bar{c})$. La fonction objective est

$$\hat{\phi}(\bar{c}) = \sum_{j \in \mathscr{D}} \hat{\nu}_j(\bar{c}). \tag{1.25}$$

Laissez $\phi(\bar{c}) = \mathbb{E}\left[\hat{\phi}(\bar{c})\right]$ être la valeur attendue où l'attente est dépassée par l'aléatoire. Fortement, le CAP est déclaré comme suit:

$$\bar{c}^* \in \underset{\bar{c} \in \mathscr{F}^{|\mathscr{D}|}}{arg\,max} \quad \phi(\bar{c}). \tag{1.26}$$

1.4.3 Le Cadre Jeux Potential Nocif

Dans les scénarios réels, les UE ne peuvent avoir que leur débit estimé qui est corrompu par le bruit. Par conséquent, nous modélisons le problème d'affectation de canal (1.26) avec des utilitaires bruyants.

Definition 8 *[CAP game] Un jeu CAP est défini par le tuple* $\hat{\mathscr{G}} := \left\{\mathscr{D}, \{X_i\}_{i \in \mathscr{D}}, \left\{\hat{U}_i\right\}_{i \in \mathscr{D}}\right\}$, *où* $\mathscr{D}$ *est un ensemble d'UE qui sont des joueurs du jeu,* $\{X_i\}_{i \in \mathscr{D}}$ *sont des ensembles d'action composés de canaux orthogonaux,* $\hat{U}_i : X \to \mathscr{R}$ *sont des fonctions d'utilité avec des attentes finies, et* $X = X_1 \times X_2 \times \ldots X_{|\mathscr{D}|}$.

Un profil d'action $a := (a_i, a_{-i})$ où $a_i \in X_i$ est l'action du joueur de i et $a_{-i} \in X_{-i}$ est l'ensemble d'action de tous les joueurs sauf le joueur i. Notez que le vecteur d'action $a \in X$ est le vecteur d'attribution de canal $\bar{c}$ et $X = \mathscr{F}^{|\mathscr{D}|}$.

Les fonctions d'utilité du jeu CAP doivent être conçues avec soin pour le modeler comme un jeu potentiellement bruyant. Nous considérons la fonction d'utilité suivante qui

Table 1.3: Parameters des Simulation.

Parameter	Variable	Valeur
Nombre de canaux orthogonaux	$\mathscr{F}$	5
Bande passante du canal	W_c	180 KHz
Fréquence porteuse	f_c	2.6 GHz
Nombre d'UE	$\lvert\mathscr{D}\rvert$	20
Puissance d'émission totale de BS	P_{BS}	46 dBm
Transmettre le pouvoir de l'UE	P_{UE}	25 dBm
Puissance de bruit additive par canal	N	$-174+10\log(W_i)$ dBm
Exposant Pathloss	η	3.5
variance shadowing	σ_{sh}	6

représente la contribution marginale du joueur à l'utilité globale qui est le taux de données de somme.

$$\hat{U}_i(a_i,a_{-i}) = \sum_{j\in\mathscr{D}(a_i)} \hat{\mathrm{v}}_j(a_i,a_{-i}) - \sum_{j\in\mathscr{D}(a_i)\setminus i} \hat{\mathrm{v}}_j(a_{-i}), \tag{1.27}$$

où $\hat{\mathrm{v}}_j$ est le taux de données mesuré du joueur j et $\mathscr{D}(a_i)\subset\mathscr{D}$ est un ensemble de joueurs dont l'action est a_i, c'est-à-dire, $\mathscr{D}(a_i)=\left\{j\in\mathscr{D}: a_j=a_i\right\}$.

Proposition 8 *Une jeux CAP* $\mathscr{G}:=\left\{\mathscr{D},\{X_i\}_{i\in\mathscr{D}},\left\{\hat{U}_i\right\}_{i\in\mathscr{D}}\right\}$ *avec function (1.27) est un jeu potentiellement bruyant avec une fonction potentielle* $\phi(a)$.

L'utilité bruyante $\hat{U}$ peut avoir une grande variance qui conduit un jeu potentiellement bruyant à avoir un écart important par rapport au jeu potentiel exact. Pour réduire la variance de l'utilité, nous utilisons un exemple de moyenne de fonction d'utilité comme ci-dessous

$$\hat{U}_i^N = \frac{1}{N}\sum_{k=1}^{N}\hat{U}_i. \tag{1.28}$$

Corollary 5 *Une jeux CAP* $\hat{\mathscr{G}}^N:=\left\{\mathscr{D},\{X_i\}_{i\in\mathscr{D}},\left\{\hat{U}_i^N\right\}_{i\in\mathscr{D}}\right\}$ *est un jeu potentiellement bruyant avec une fonction potentielle* $\phi(a)$.

Dans le reste du section, nous considérons le jeu de CAP potentiel bruyant $\hat{\mathscr{G}}^N$. Nous appliquons l'algorithme log-linéaire binaire proposé (BLLA) dans les jeux potentiels bruyants pour la résolution distributive.

1.4.4 Simulations

Dans cette section, nous présentons des résultats de simulation en tenant compte des paramètres standard du système sans fil affichés dans Table 1.3. Nous considérons qu'une

BS est située au centre d'une région de rayon de 200 m. Parmi les 20 UE, il y a 5 UEC. Les UEC ont des canaux dédiés et aucun deux UEC n'est sur le même canal. Ces UEC servent de joueurs passifs du jeu parce qu'ils ne changent pas leur chaîne. Les récepteurs des émetteurs UED sont situés autour d'eux uniformément aléatoires sur une région de rayon de 20 m. Les UED apprennent leur chaîne avec l'aide de BS et sont donc les acteurs actifs du jeu.

Les variations de la disparition de Rayleigh au fil du temps sont considérées comme la composante du bruit pour toutes les simulations. Ces variations apparaissent dans les débits de données des UE. Les débits de données sont délimités car le SINR est délimité entre $\gamma_{\min}$ et $\gamma_{\max}$. Nous n'assumons que le bruit borné dans les simulations où l'intervalle de bruit $\ell = 1$ en raison des utilitaires normalisés. En outre, le bruit additif Gaussien blanc avec puissance N est considéré.

Dans cette sous-section, nous montrons les simulations pour la liaison descendante du système. Dans Fig. 1.6, la convergence vers le taux de données de somme maximal de BLLA est affichée avec une température fixe $tau = 0.1$ et une température décroissante $\tau(t) = 0.1/\log(1+t)$ [2]. Le nombre d'échantillons est calculé selon (1.12) du théorème 2 pour $\tau = 0.1$ et $\xi = 10^{-5}$. BLLA atteint le taux de données de somme maximal avec des températures fixes et décroissantes. Cependant, il présente plus de variations pour la température fixe. Pour diminuer la température, la probabilité de rester au maximum est plus élevée, ce qui est évident à partir de Fig. 1.6.

Nous étudions maintenant l'effet du nombre de canaux dans Fig. 1.7. Ce graphique montre le taux moyen de données de somme obtenu à partir de BLLA au bout de 500 itérations moyennes sur 1000 réalisations. Nous voyons que le taux de données de somme augmente à mesure que le nombre de canaux augmente. Ceci est intuitif car l'assignation de canal optimale a une interférence plus faible par canal. Comme le montre clairement la figure, BLLA suit correctement ce phénomène.

Nous étudions également les performances de BLLA en faisant varier le nombre d'UE dans Fig. 1.8 pour 10 canaux orthogonaux et 10 UEC. Comme précédemment, le taux de données de somme est obtenu à partir de BLLA au bout de 500 itérations moyennes sur 1000 réalisations.

[2] Notez que $\tau(t) = 0.1/\log(1+t)$ fonctionne bien bien qu'il soit plus petit que celui donné par Theorem **??**. La raison en est que la hauteur du maximum local le plus élevé est inférieure au maximum global [7]. Nous considérons que cette hauteur est de 10 % du maximum global, ce qui est raisonnable.

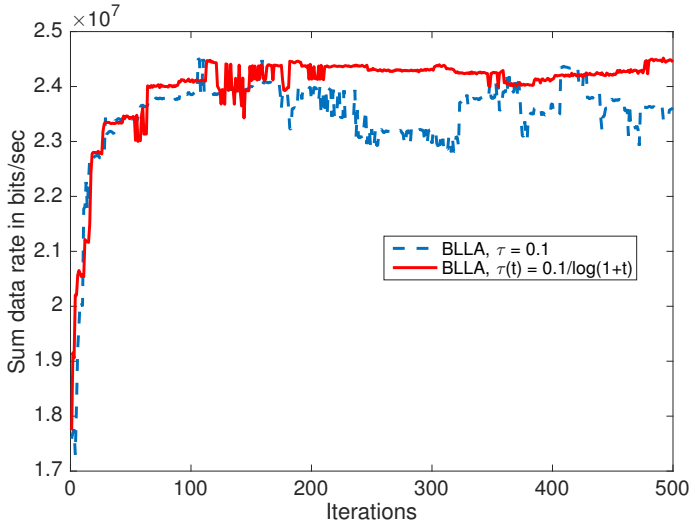


Figure 1.6: [Convergence de BLLA en liaison descendante pour température fixe et température décroissante]: On constate que la température décroissante entraîne une convergence plus lisse par rapport à celle fixée.

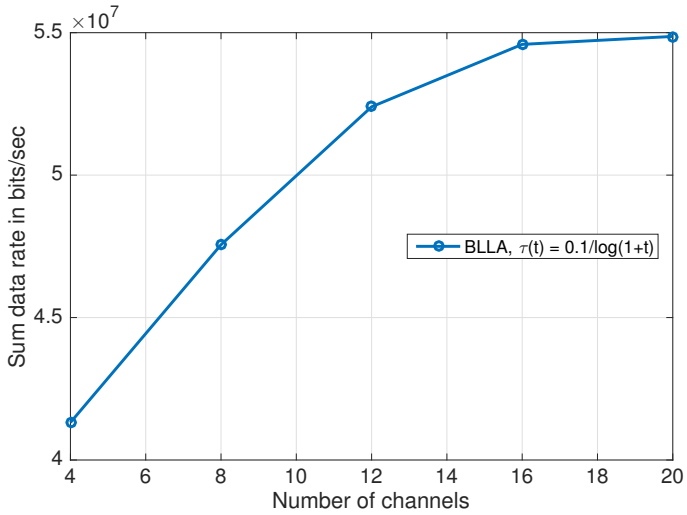


Figure 1.7: [Effet d'un certain nombre de canaux sur le taux de données de somme]: Le nombre d'UE est fixé à 20. Nous voyons que le taux de somme des données augmente avec le nombre de canaux puisque BLLA affecte des canaux de manière optimale conduisant à une interférence plus faible par canal.

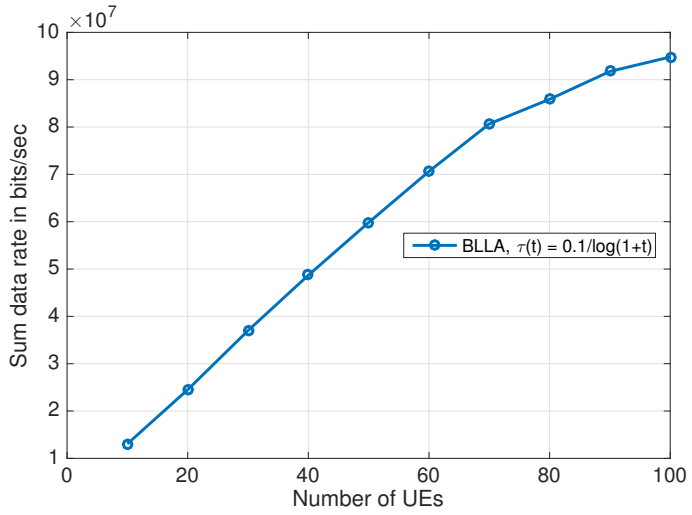


Figure 1.8: [Effet du nombre d'UE sur le taux de données de somme]: le nombre de canaux est fixé à 10. Nous voyons que le taux de somme des données augmente linéairement jusqu'à 60 EU UE car BLLA parvient à affecter les canaux de façon optimale et à maintenir une faible interférence. Pour plus d'UE de 60, l'interférence par canal augmente de façon significative les effets du taux de données de somme.

1.5 Algorithme glouton distribué pour la maximisation sous modulaire

Dans ce section, nous considérons la maximisation distribuée d'une fonction submodulaire, qui est une fonction définie qui a la propriété que la différence de fonction diminue avec la taille croissante d'un ensemble donné lorsqu'un élément lui est ajouté.

L'optimisation des fonctions submodulaires est un sujet bien étudié en raison de son application dans de nombreux problèmes d'ingénierie communs. Les exemples comprennent collecte d'informations [42], maximiser l'influence(marketing viral)dans les réseaux sociaux [43], segmentation d'image dans le traitement d'image [44], Détection d'objets multiples dans la vision par ordinateur [45], résumés des documents [46], placement de capteur [47, 48], et allocation de ressources dans les systèmes multi-agents [48]. Pour chaque cas, le problème peut être formulé comme l'optimisation d'une fonction submodulaire.

Ce travail se rapporte plus étroitement au travail effectué dans [49] dans l'évaluation des contraintes de partage d'informations. Leur travail modélise les structures de communication sous forme de graphiques et démontre comment les performances peuvent se dégrader pour certains graphiques. Par exemple, si les agents sont répartis en groupes qui ne transmettent que parmi eux, la limite inférieure de performance se dégrade d'un facteur

proportionnel au nombre de groupes. Cependant, deux questions restent ouvertes:

1. Pour toute structure de communication, quelles sont les garanties de performance?
2. Quelles structures de communication rendent les meilleures garanties de performance?

1.5.1 Model

Ce travail se concentre sur la conception d'algorithmes distribués pour obtenir des solutions souhaitables pour la maximisation submodulaire. À cette fin, laissez S un ensemble d'éléments et $f : 2^S \to \mathbb{R}_{\geq 0}$ ont les propriétés suivantes:

- *Submodulaire*:For $A \subseteq B \subset S$ and $x \in S \setminus B$, ce qui suit:

$$f(A \cup \{x\}) - f(A) \geq f(B \cup \{x\}) - f(B). \tag{1.29}$$

- *Monotone*:For $A \subseteq B$, $f(A) \leq f(B)$.
- *Normalized*:$f(\emptyset) = 0$.

Nous disons que f is *modular*if (1.29)est une égalité.

Dans de nombreuses applications, la maximisation de ces f peut être formulée comme la compilation des décisions prises par les agents, chacun maximisant une utilité locale. Supposons que nous ayons des agents marqués $1, ..., n$ et une famille de jeux correspondante $X_1, ..., X_n$, où $X_i \subseteq 2^S$. Ici, nous définissons $X = X_1 \times \cdots \times X_n$. À des fins de notation, si A est un ensemble d'agents, laissez $x_A = \bigcup_A x_i$, où $x_i \in X_i$ et $i \in A$. Nous utilisons également la notation que $x_{a:b} = \bigcup_{i=a}^{b} x_i$ pour $b \geq a$. Enfin, dans ce qui suit, nous définissons $x_1^*, ..., x_n^*$ comme l'ensemble optimal des décisions pour chaque agent. Dans ce contexte, le problème de la maximisation submodulaire peut être formulé comme

$$f(x_{1:n}^*) = \max_{x_1 \in X_1, ..., x_n \in X_n} f(x_{1:n}). \tag{1.30}$$

Ce type de contrainte où nous choisissons parmi une famille de sous-ensembles X_i équivaut à une partition de contrainte matroid [49]. Chaque X_i peut être considéré comme l'espace d'action de l'agent i.

Comme mentionné dans la section précédente, le problème de maximisation dans (1.30) peut être approché à l'aide d'un algorithme gourmand distribué. Étant donné que l'algorithme

exige que les agents prennent des décisions séquentiellement, sans perte de généralité, nous imposons une commande aux agents en fonction de leurs étiquettes, c'est-à-dire que l'agent 1 choisit d'abord, l'agent 2 choisit le second, etc. L'agent i choisit $x_i \in X_i$ selon la règle suivante:

$$x_i \in arg\,max_{\tilde{x}_i \in X_i} f(\tilde{x}_i \cup x_{1:i-1}), \tag{1.31}$$

où $x_{1:0}$ est défini comme $\emptyset$. Notez qu'il pourrait y avoir plusieurs x_i à choisir dans $arg\,max$, alors définissez $\mathscr{X} \subseteq X$ pour être l'ensemble de tous les choix possibles si les agents choisissent selon (1.31). Il est bien connu dans la littérature que, pour tout sous-modèle f, un ensemble d'espaces d'action X, et n'importe quel ordre d'agents, la qualité de la solution résultante $x = x_{1:n}$ dérivée de la algorithme gourmand distribué par rapport à la solution optimale $x^* = x^*_{1:n}$ satisfait

$$\gamma(f,X) := \frac{\inf_{x \in \mathscr{X}} f(x)}{f(x^*)} \geq \frac{1}{2} \tag{1.32}$$

ce qui signifie que la solution se situe à moins de 50% de l'optimale [50]. Nous disons que $\gamma(f,X)$ est l'efficacité de la solution x. Pour les classes spéciales de fonctions submodulaires et les contraintes supplémentaires sur $X_1,...,X_n$, [50] montre également que $\gamma(f,X) \geq 1-1/e$. Notez que cet algorithme gourmand est différent par rapport au meilleur algorithme de réponse, dans lequel l'agent actuel doit connaître l'action de tous les agents.

Notez que dans (1.31), agent i doit avoir accès aux décisions de tous les agents précédents. Toutefois, cela peut ne pas se produire dans certaines applications du monde réel. Par conséquent, une version plus généralisée de l'algorithme greedy distribué est l'endroit où agent i fait son choix en utilisant la règle suivante:

$$x_i \in arg\,max_{\tilde{x}_i \in X_i} f(\tilde{x}_i \cup x_{\mathscr{N}_i}), \tag{1.33}$$

où $\mathscr{N}_i \subseteq \{1,...,i-1\}$. Les ensembles $\mathscr{N}_i$, $i = 1,...,n$ sont les ensembles qui représentent la structure de communication - en d'autres termes, $\mathscr{N}_i$ est l'ensemble d'agents dont les choix sont observés par agent i lors de la prise de sa décision. Il est utile de visualiser cette structure de communication sous la forme d'un graphe dirigé $G = (V,E)$, où V est un ensemble de sommets et $E \subseteq V \times V$ est un ensemble d'arêtes entre les sommets. Dans ce scénario, chaque sommet est un agent et chaque bord (j,i) implique $j \in \mathscr{N}_i$, c.-à-d. $\mathscr{N}_i$ est l'ensemble des voisins du sommet i. Comme il y a une ordonnance imposée sur les

sommets, et les agents choisissent séquentiellement, l'ensemble

$$\mathscr{G} = \{G = (V,E) : (i,j) \in E \implies i < j\} \tag{1.34}$$

est l'ensemble des graphiques admissibles qui correspondent à une structure de communication. Dans ce contexte, nous définissons $\mathscr{X} \subseteq X$ pour être l'ensemble de tous les choix possibles si les agents choisissent selon (1.33)et mesurent encore l'efficacité de la solution x en fonction de la règle de décision (1.33)avec

$$\gamma(f,X,G) := \frac{\inf_{x\in\mathscr{X}} f(x)}{f(x^*)}, \tag{1.35}$$

où $G \in \mathscr{G}$ est défini par les ensembles $\mathscr{N}_i$.

L'objectif de ce travail sera d'identifier les garanties d'efficacité associées à cette version plus généralisée de l'algorithme gourmand distribué pour toute fonction submodulaire monotone normalisée. À cette fin, laissez $\mathscr{F}$ être l'ensemble de ces fonctions, où chaque $f \in \mathscr{F}$ est défini sur un élément défini S_f. Nous définissons maintenant

$$\gamma(G) = \inf_{f\in\mathscr{F}} \inf_{X:X_i\subseteq 2^{S_f}} \gamma(f,X,G) \tag{1.36}$$

Par mots, $\gamma(G)$ est l'efficacité la plus défavorable pour f et la famille de jeux $X_1,...,X_n$ tel que défini ci-dessus, étant donné que la structure de communication parmi les agents est représentée par G.

1.5.2 Limites inférieures et supérieures sur Efficience $\gamma(G)$

Maintenant, nous présentons les limites inférieures et supérieures sur $\gamma(G)$ dans les termes du nombre de couverture de la clique $k(G)$ et le nombre d'indépendance $\alpha(G)$, respectivement.

Theorem 9 *Laissez $G \in \mathscr{G}$. Alors*

$$\frac{1}{\alpha(G)} \geq \gamma(G) \geq \frac{1}{k(G)+1}. \tag{1.37}$$

Comme indiqué, la limite inférieure présentée dans Theorem 9 est supérieure ou égale à ces limites inférieures montrées dans [49].

1.5.3 Graphique de communication presque optimisé

Les limites de $\gamma(G)$ données ci-dessus dictent une approche presque optimale de la conception de la structure de la communication. À l'aide de ces résultats, nous décrivons comment créer des structures de graphiques qui correspondent au plus élevé $\gamma(G)$. En d'autres termes, compte tenu de n vértices et de m, nous montrons où placer les bords afin de maximiser $\gamma(G)$. Nous commençons par définir la notation suivante:

- Soit $\mathscr{G}_{m,n} := \{G = (V,E) \in \mathscr{G} : |V| = n, |E| = m\}$.
- Indiquez
$$G^*_{m,n} \in arg\,max_{G \in \mathscr{G}_{m,n}} \gamma(G). \tag{1.38}$$
Notez que $G^*_{m,n}$ n'est pas nécessairement unique pour un m et n.
- Pour un graphique $G = (V,E)$, son *complément* est $\bar{G} = (V,\bar{E})$, où $(i,j) \in E \iff (i,j) \notin \bar{E}$.

Theorem 10 *Pour tous les nombres entiers $m \geq 0$ et $n > 0$ tels que $m \leq \frac{1}{2}n(n-1)$,*

$$\frac{1}{r^*} \geq \gamma(G^*_{m,n}) \geq \gamma(\hat{T}(n,r^*)) = \frac{1}{r^*+1} \tag{1.39}$$

où

$$r^* = \left\lceil \frac{n^2}{2m+n} \right\rceil \tag{1.40}$$

et $\hat{T}(n,r) \in \mathscr{G}$ pour tout entier positiver r est un graphique construit avec l'algorithme suivant:

1. *Partitionnez les sommets dans r différents ensembles $C_1,...,C_r$ tel que $|C_i|$ et $|C_j|$ diffèrent de pas plus de 1 pour tous $i,j \in \{1,...,r\}$. En d'autres termes, tous les jeux de la partition sont aussi proches que possible.*
2. *Créez des bords entre tous les noeuds de chaque ensemble.*

Conclusion

L'objectif de la thèse est d'analyser les algorithmes d'apprentissage distribués et de les appliquer aux réseaux de communication sans fil. Nous étudions deux algorithmes pour les jeux à potentiel immédiat: Log-Linear Learning Algorithm (LLA) et BLA LLA (BLLA) pour les paramètres d'information complète et partielle, respectivement. Nous proposons de nouvelles règles pour le calcul de la résistance des arbres d'une chaîne de Markov Perturbed (PMC). Ces règles nous permettent de prouver la convergence des algorithmes. Nous prouvons que les algorithmes convergent vers le minimum global de la fonction potentielle sous une certaine condition, qui capture le compromis entre la qualité de la solution obtenue et la distance à un jeu potentiel exact. Ensuite, nous analysons la dynamique de BLLA dans les jeux potentiellement bruyants, qui tiennent compte des utilités bruyantes.

Nous appliquons les resultants dans des jeux potentiellement potentiels pour l'équilibrage de charge parmi les stations de base dans des réseaux cellulaires hétérogènes en utilisant le biais de l'extension de portée de la cellule (CRE) pour l'association des utilisateurs et la sous-châssis presque vierge (ABS) pour la gestion des interférences. Nous considérons une fonction d'objectif α -fairness avec des contraintes de panne et de charge. Nous obtenons des conditions pratiques en termes de paramètres système qui garantissent la convergence de LLA et BLLA. Nous effectuons des simulations complètes qui montrent que les algorithmes proposés atteignent l'équilibrage de charge désiré. On constate que le modèle de jeu à potentiel proche apparaît naturellement dans un scénario de décoloration des ombres. Nous montrons que l'ABS contribue à la contrainte de panne et à un meilleur équilibrage de charge. Nous développons également un nouvel algorithme d'apprentissage pour trouver automatiquement la meilleure température τ à utiliser pour LLA et BLLA.

Ensuite, nous appliquons BLLA dans des jeux à potentiel bruyant à problème d'assignation de canal (CAP) dans les réseaux sans fil Device-to-Device (D2D), lorsque l'estimation du débit est corrompue par le bruit. En utilisant le nombre suffisant d'échantillons à partir des résultats de convergence de BLLA, l'affectation optimale des canaux est obtenue.

Enfin, nous abordons un problème généralisé de maximisation submodulaire à l'aide de jeux avec des joueurs ayant des informations limitées ou inexistantes sur leur quartier. Nous donnons des limites inférieures et supérieures sur la performance la plus défavorable de l'algorithme gourmand pour un graphique d'information donné. Ensuite, nous déterminons le meilleur graphique d'information qui donne les meilleures performances qu'un concepteur de système peut réaliser avec un nombre fixe d'agents et des bordures d'information.

Bibliography

[1] J. Mo and J. Walrand, "Fair end-to-end window-based congestion control," *IEEE/ACM Trans. Netw.*, vol. 8, no. 5, pp. 556–567, 2000.

[2] J.-Y. Audibert and S. Bubeck, "Best arm identification in multi-armed bandits," in *COLT-23th Conference on Learning Theory*, pp. 13–p, 2010.

[3] D. Monderer and L. S. Shapley, "Potential Games," *Games and Economic Behavior*, vol. 14, pp. 124–143, May 1996.

[4] O. Candogan, A. Ozdaglar, and P. A. Parrilo, "Learning in near-potential games," in *IEEE Conf. Decision Control and European Control Conference*, pp. 2428–2433, 2011.

[5] J. R. Marden and J. S. Shamma, "Revisiting log-linear learning: Asynchrony, completeness and a payoff-based implementation," *Games and Economic Behaviour*, vol. 75, pp. 788–808, July 2012.

[6] D. S. Leslie and J. R. Marden, "Equilibrium selection in potential games with noisy rewards," in *Proc. IEEE Network Games, Control and Optimization (NetGCooP)*, pp. 1–4, 2011.

[7] B. Hajek, "Cooling schedules for optimal annealing," *Mathematics of operations research*, vol. 13, no. 2, pp. 311–329, 1988.

[8] J. Andrews, S. Singh, Q. Ye, X. Lin, and H. Dhillon, "An overview of load balancing in hetnets: old myths and open problems," *IEEE Wireless Commun.*, vol. 21, pp. 18–25, April 2014.

[9] O. K. Tonguz and E. Yanmaz, "The mathematical theory of dynamic load balancing in cellular networks," *IEEE Trans. Mobile Comput.*, vol. 7, no. 12, pp. 1504–1518, 2008.

[10] H. Kim, G. de Veciana, X. Yang, and M. Venkatachalam, "Distributed alpha-optimal user association and cell load balancing in wireless networks," *IEEE/ACM Trans. Netw.*, vol. 20, pp. 177–190, Feb 2012.

[11] F. Xu, C. C. Tan, Q. Li, G. Yan, and J. Wu, "Designing a practical access point association protocol," in *Proc. IEEE INFOCOM*, pp. 1–9, 2010.

[12] D. Niyato and E. Hossain, "Dynamics of network selection in heterogeneous wireless networks: An evolutionary game approach," *IEEE Trans. Veh. Technol.*, vol. 58, pp. 2008–2017, May 2009.

[13] E. Aryafar, A. Keshavarz-Haddad, M. Wang, and M. Chiang, "RAT selection games in HetNets," in *Proc. IEEE INFOCOM*, pp. 998–1006, 2013.

[14] T. Kudo and T. Ohtsuki, "Cell range expansion using distributed Q-learning in heterogeneous networks," *EURASIP J. Wireless Commun. Netw.*, pp. 1–10, Mar. 2013.

[15] 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 Commun.*, vol. 18, no. 3, pp. 10–21, 2011.

[16] J. Oh and Y. Han, "Cell selection for range expansion with almost blank subframe in heterogeneous networks," in *Proc. PIMRC*, pp. 653–657, Sept 2012.

[17] S. Singh, X. Zhang, and J. Andrews, "Joint rate and sinr coverage analysis for decoupled uplink-downlink biased cell associations in hetnets," *IEEE Trans. Wireless Commun.*, vol. 14, pp. 5360–5373, Oct 2015.

[18] R. Madan, J. Borran, A. Sampath, N. Bhushan, A. Khandekar, and T. Ji, "Cell association and interference coordination in heterogeneous LTE-A cellular networks," *IEEE J. Sel. Areas Commun.*, vol. 28, pp. 1479–1489, Dec. 2010.

[19] K. Okino, T. Nakayama, C. Yamazaki, H. Sato, and Y. Kusano, "Pico cell range expansion with interference mitigation toward lte-advanced heterogeneous networks," in *ICC Workshop*, pp. 1–5, June 2011.

[20] M. Shirakabe, A. Morimoto, and N. Miki, "Performance evaluation of inter-cell interference coordination and cell range expansion in heterogeneous networks for LTE-advanced downlink," in *ISWCS*, pp. 844–848, Nov. 2011.

[21] M. Vajapeyam, A. Damnjanovic, J. Montojo, T. Ji, Y. Wei, and D. Malladi, "Downlink FTP performance of heterogeneous networks for LTE-advanced," in *ICC Workshop*, pp. 1–5, June 2011.

[22] I. Guvenc, "Capacity and fairness analysis of heterogeneous networks with range expansion and interference coordination," *IEEE Commun. Lett.*, vol. 15, pp. 1084–1087, Oct. 2011.

[23] A. Morimoto, N. Miki, H. Ishii, and D. Nishikawa, "Investigation on transmission power control in heterogeneous network employing cell range expansion for LTE-advanced uplink," in *European Wireless Conf.*, pp. 1–6, Apr. 2012.

[24] K. Kitagawa, T. Komine, T. Yamamoto, and S. Konishi, "Performance evaluation of handover in LTE-advanced systems with pico cell range expansion," in *Proc. PIMRC*, pp. 1071–1076, Sep. 2012.

[25] M. Eguizabal and A. Hernandez, "Interference management and cell range expansion analysis for LTE picocell deployments," in *Proc. PIMRC*, pp. 1592–1597, Sept 2013.

[26] T. Bonald and L. Massoulié, "Impact of fairness on internet performance," in *ACM SIGMETRICS Performance Evaluation Review*, vol. 29, pp. 82–91, 2001.

[27] Y. Xu, J. Wang, Q. Wu, A. Anpalagan, and Y.-D. Yao, "Opportunistic spectrum access in cognitive radio networks: Global optimization using local interaction games," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 6, pp. 180–194, April 2012.

[28] K. Yamamoto, "A comprehensive survey of potential game approaches to wireless networks," *IEICE Transactions on Communications*, vol. 98, no. 9, pp. 1804–1823, 2015.

[29] 3GPP, "Technical specification group radio access network; evolved universal terrestrial radio access (eutra); further advancements for e-utra physical layer aspects," TR 36.814, 3GPP, Mar. 2010.

[30] A. Asadi, Q. Wang, and V. Mancuso, "A survey on device-to-device communication in cellular networks," *IEEE Commun. Surveys & Tutorials*, vol. 16, no. 4, pp. 1801–1819, 2014.

[31] L. Song, D. Niyato, Z. Han, and E. Hossain, "Game-theoretic resource allocation methods for device-to-device communication," *IEEE Commun. Lett.*, vol. 21, no. 3, pp. 136–144, 2014.

[32] Q. Ye, M. Al-Shalash, C. Caramanis, and J. G. Andrews, "Distributed resource allocation in device-to-device enhanced cellular networks," *IEEE Trans. Commun.*, vol. 63, no. 2, pp. 441–454, 2015.

[33] Y. Li, D. Jin, J. Yuan, and Z. Han, "Coalitional games for resource allocation in the device-to-device uplink underlaying cellular networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 7, pp. 3965–3977, 2014.

[34] H. Chen, D. Wu, and Y. Cai, "Coalition formation game for green resource management in d2d communications," *IEEE Commun. Lett.*, vol. 18, no. 8, pp. 1395–1398, 2014.

[35] Y. Cai, H. Chen, D. Wu, W. Yang, and L. Zhou, "A distributed resource management scheme for d2d communications based on coalition formation game," in *Proc. ICC*, pp. 355–359, IEEE, 2014.

[36] B.-Y. Huang, S.-T. Su, C.-Y. Wang, C.-W. Yeh, and H.-Y. Wei, "Resource allocation in d2d communication-a game theoretic approach," in *Proc. ICC*, pp. 483–488, IEEE, 2014.

[37] C. Xu, L. Song, Z. Han, Q. Zhao, X. Wang, X. Cheng, and B. Jiao, "Efficiency resource allocation for device-to-device underlay communication systems: a reverse iterative combinatorial auction based approach," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 9, pp. 348–358, 2013.

[38] R. Wang, J. Zhang, S. Song, and K. B. Letaief, "Optimal qos-aware channel assignment in d2d communications with partial csi," *IEEE Trans. Wireless Commun.*, vol. 15, no. 11, pp. 7594–7609, 2016.

[39] S. Maghsudi and S. Stańczak, "Hybrid centralized–distributed resource allocation for device-to-device communication underlaying cellular networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 4, pp. 2481–2495, 2016.

[40] C. Gao, X. Sheng, J. Tang, W. Zhang, S. Zou, and M. Guizani, "Joint mode selection, channel allocation and power assignment for green device-to-device communications," in *Proc. ICC*, pp. 178–183, 2014.

[41] M. N. Tehrani, M. Uysal, and H. Yanikomeroglu, "Device-to-device communication in 5g cellular networks: challenges, solutions, and future directions," *IEEE Commun. Mag.*, vol. 52, no. 5, pp. 86–92, 2014.

[42] A. Krause and C. Guestrin, "Near-optimal observation selection using submodular functions," in *AAAI*, vol. 7, pp. 1650–1654, 2007.

[43] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 137–146, ACM, 2003.

[44] P. Kohli, M. P. Kumar, and P. H. Torr, "P^3 & beyond: Move making algorithms for solving higher order functions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 9, pp. 1645–1656, 2009.

[45] O. Barinova, V. Lempitsky, and P. Kholi, "On detection of multiple object instances using hough transforms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 9, pp. 1773–1784, 2012.

[46] H. Lin and J. Bilmes, "A class of submodular functions for document summarization," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pp. 510–520, Association for Computational Linguistics, 2011.

[47] A. Krause, R. Rajagopal, A. Gupta, and C. Guestrin, "Simultaneous placement and scheduling of sensors," in *Proceedings of the 2009 international Conference on information Processing in Sensor Networks*, pp. 181–192, IEEE Computer Society, 2009.

[48] J. R. Marden, "The role of information in distributed resource allocation," *IEEE Transactions on Control of Network Systems*, 2016.

[49] B. Gharesifard and S. L. Smith, "On distributed submodular maximization with limited information," in *American Control Conference (ACC)*, pp. 1048–1053, IEEE, 2016.

[50] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, "An analysis of approximations for maximizing submodular set functions I," *Mathematical Programming*, vol. 14, no. 1, pp. 265–294, 1978.

ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY

TELECOM ParisTech

2017-ENST-0030

EDITE - ED 130

Doctorate ParisTech

Thesis

for obtaining the degree of doctorate by

TELECOM ParisTech

Speciality « Computer Science & Networks »

presented and publicly defended by

Mohammed Shabbir ALI

27th June 2017

Title

Distributed Learning in Games for Wireless Networks

Directeur de thèse : **Marceau COUPECHOUX**
Co-encadrement de la thèse : **Pierre COUCHENEY**

Jury

M. Bruno GAUJAL, Directeur de Recherches, INRIA	Rapporteur
M. Ekram HOSSAIN, Professeur, University of Manitoba	Rapporteur
M. Eitan ALTMAN, Directeur de Recherches, INRIA	Examinateur
Mme. Johanne COHEN, Directeur de Recherches, CNRS	Examinateur
M. Jean-Marie GORCE, Professeur, INSA-Lyon	Examinateur
M. Olivier MARCE, Ingenieur de Recherche, Nokia Bell Labs	Examinateur

TELECOM ParisTech
école de l'Institut Mines-Télécom - membre de ParisTech
46 rue Barrault 75013 Paris - (+33) 1 45 81 77 77 - www.telecom-paristech.fr

THÈSE

Abstract

In this thesis, we address the issue of optimal design and management of wireless networks, which have become crucial with the increasing demand for data. Centralized approaches are becoming infeasible, while distributed learning in games is a promising distributed approach. Particularly, potential games are attractive as the players of the game can distributively optimize a potential function. We extend this notion to near and noisy potential games, and study the convergence of learning algorithms in these frameworks. We then apply the obtained theoretical results to wireless networks. In a first part, we prove that under certain conditions Log-linear Learning Algorithm (LLA) and Binary LLA (BLLA) converge to the global minimum of the potential function in near potential g ames. We then prove the convergence of BLLA in noisy-potential games for fixed and decreasing temperature. We provide a sufficient number of estimation samples that guarantees the convergence for both bounded and unbounded noise. A key enabler for analyzing the convergence of the proposed algorithms is the resistance of trees in a Perturbed Markov Chain (PMC). We develop new rules to simplify the calculation of the resistance of trees. In a second part, we apply our results to practical problems in wireless networks. We first consider load balancing in heterogeneous cellular networks and propose a distributed algorithm to minimize an α-fairness objective function that captures various network performance and fairness criteria. The near-potential game approach allows us to decrease the size of base station's neighborhood, with which it has to exchange information. In this context, we also propose a new annealing schedule for LLA adapted to finite time horizons. Next, we apply BLLA to Channel Assignment Problem (CAP) in Device-to-Device (D2D) networks. Because of the rate estimation noise, the noisy-potential game framework naturally arises. Moving on from potential games, we address a general constrained submodular maximization problem. We model it as a finite game with players having limited or no information about their neighborhood. We characterize the performance of the greedy algorithm and we provide the best information graph for a given number of players and information edges.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

World's sustainability depends on the control engineering systems like communication network systems, electrical power systems, transport systems, etc. that supply the needs of the world. These systems are growing exponentially with the ever growing demands of the increasing population. Efficient design and management of these systems is essential. Centralized approaches for this purpose are becoming infeasible due to the large size and huge complexity. Distributed approaches seem to be the only promising alternatives. The challenge is to design distributed algorithms that can achieve the global optimal performance.

Game theory has become attractive as it provides a tool for distributed approach. Many engineering problems can be modeled as a game. A game consists of a set of players or agents, their action sets, and their utilities or cost functions. Multiple players interact with each other to arrive at some equilibrium state. One notion of equilibrium is Nash Equilibrium (NE), in which no player can gain by unilateral deviation. The players are generally assumed to be rational i.e., with the amount of information the players have they choose an action that maximizes their utility. This myopic behavior of the players may lead to suboptimal NE that is inefficient. Thus, an important challenge is how to achieve the optimal NE. Potential games are promising to address this challenge. In potential games, the players can distributively achieve the global optimal of the potential function. Potential games have many important properties. They are guaranteed to have an NE. All the NEs of a potential game are contained in the set of optima of the potential function. However, it is difficult to have potential games in many problems. Near-potential games expand the scope of potential games to a wider class of problems. They relax the strict constraints of the potential games and at the same time share similar properties of distributed nature.

Noisy-potential games further expands the scope of potential games to a class of stochastic optimization problems by taking into account the noisy utilities of the players. In these games, the goal is to reach the optimal average value of the potential function. However, to obtain the above potential games the utilities or cost functions of the players have to be carefully designed, which may not be possible in some cases. This may be due to the lack of information at the player. However, if the objective function has some properties such as submodular property then it can be exploited. A submodular function is a set function that has the property that the difference of the function decreases with the increasing size of a given set when an element is added to it. A distributed greedy algorithm can be used for maximizing a submodular function. The greedy algorithm can guarantee some worst case performance.

In distributed learning, the players distributively learn their optimal actions. Only local neighborhood information may be available to the players. This allows solving problems efficiently without much overhead of information exchange. Therefore, distributed learning finds applications in many engineering systems such as wireless communication networks, wireless device-to-device (D2D) networks, and wireless sensor networks. Distributed learning can be applied to optimization problems that can be modeled as potential games. For this purpose, the objective function of the problem is turned into the potential function of the game. The optimal of the objective will then be the optimal NE of the game. For a different class of potential functions, different distributed learning algorithms are required. For the class of convex or concave potential functions, a simple greedy or best response algorithm can achieve the optimal NE. However, in general, convexity does not occur in many problems. The best response algorithm does not guarantee the optimal as it may get stuck into a local minimum of the non-convex function. Here, the challenge is to design new distributed algorithms that can escape the local minimum and achieve the global minimum. Moreover, if the potential function is corrupted by some sort of noise like estimation noise then the problem becomes even more challenging.

Distributed learning is promising for wireless communication networks. Wireless communication has become essential in our life due to increased dependency on the information exchange between people. Ease of access and mobility make the cellular networks attractive. Distributed learning helps in the design and management of wireless communication networks. Particularly, distributed learning is essential for cellular wireless communication networks that have base stations serving the mobile users. To supply the increasing demand of mobile communication data rates the networks are increasing in

size and complexity. A centralized control of wireless networks requires a huge amount of information exchange and computation power. Also, latency in information exchange is a critical issue to be addressed. Therefore, it is critical to design distributed learning algorithms that require low information exchange and that achieve the global optimal performance.

Communication networks are becoming heterogeneous containing different types of base stations in large numbers. Macro base stations with a high transmit power provide the coverage for a large region. Several small base stations like micro, pico and Femto base stations are deployed at the hot-spots of the traffic locations of the coverage region to provide high traffic demands. Small base stations are deployed in an ad-hoc manner in order to increase the reuse of the frequency spectrum, where macro and small base stations use the same frequency spectrum. However, reusing the same channel cause the signals of base stations to interfere with each other. This is called co-channel interference. Also, due to the heterogeneity of the base stations loads of the base stations are not balanced. For example, macro base stations are highly loaded as many users get associated with them. Whereas, the resources of the small base stations are not well utilized due small coverage regions. An important challenge is to minimize co-channel interference and associate mobile users to base stations in a way to maximize the performance of the networks. Different performance measures can be considered such as sum data rate maximization, proportional fairness, minimum average delay of the network, and a min-max load of the base stations. One of the important techniques proposed in the literature for user association is Cell Range Extension (CRE) bias association along with Almost Blank Subframe (ABS) for interference management. In CRE bias association technique, users are biased to associate with the small base stations to better utilize their resources. While using ABS some base stations limit their transmit power in some slots to reduce the interference. If the interaction of the base stations to associate the users are modeled as a potential game or as a near-potential game then the distributed learning can be applied. Particularly, in these networks, near-potential games are important to limit the size of the neighborhood of the base stations.

D2D networks are becoming important as they further increase the spectrum reuse. In D2D networks, the users communicate with each other without or with limited help from the base stations. The users reuse the same channel as the base stations increasing the spectrum reuse. However, co-channel interference also further increases causing adverse effects to the network. The interference can be managed if the channels are allocated

carefully. For example, the users who are geographically near should not be allocated the same channel. The Channel Assignment Problem (CAP) is a standard problem in all wireless network. Solving CAP in a centralized way is a very difficult problem. One reason is that it requires the perfect channel information of all links of the network. Even if this information is available still it is a difficult problem. It becomes even more difficult with the imperfect channel information. Distributed learning can help to address the CAP with the imperfect channel information. This can be done by modeling the CAP with imperfect information as a noisy-potential game.

In summary distributed learning can be applied to wide variety of problems where centralized approaches become impractical. The focus of the thesis is to study distributed learning algorithms in games. Then, the algorithms are applied to solve problems of wireless communications networks. In the following, we first present the main contributions of the thesis then we give an overview of the thesis. Also, a related literature on distributed learning approaches is presented.

1.1 Main Contributions of Thesis

We summarize the main contributions of thesis below.

1. *Distributed Learning Algorithms in Near-Potential Games:*
   We study two distributed learning algorithms for near-potential games in Section 2.2.1 of Chapter 2. First, Log-linear Learning Algorithm (LLA) for full information where the current player has full information of all his actions. Second, a Binary Log-linear Learning Algorithm (BLLA) in partial information setting where the current player has information of only two of his actions. The key idea of both of these algorithms is to play suboptimal actions sometimes to escape a local minimum and reach the global minimum. The probability of playing suboptimal actions is governed by a temperature parameter τ. We prove that under certain conditions LLA and BLLA converge to the global minimum of the potential function.

2. *Distributed Learning Algorithm in Noisy-Potential Games:*
   We introduce noisy-potential games in Section 2.2.2 of Chapter 2, to take into account the noise in the utilities of the players. The goal is to achieve the optimal average value of the potential function. Noisy-potential games can be used to model the noise of a class of SOPs whose objective function has a noise component. Then, we study a

distributed Binary Log-linear Learning Algorithm (BLLA) for noisy-potential games. In this BLLA for noisy-potential games, the players play their actions multiple times to decrease the variance of the noise so as to obtain the convergence. The number of times an action is played depends on whether the noise is bounded or unbounded. The convergence of BLLA in a noisy-potential game is proved for fixed temperature and decreasing temperature. We provide a sufficient number of estimation samples that guarantees the convergence for both the cases of bounded noise and unbounded noise.

3. *Resistance Rules for Analyzing Learning Algorithms:*
   We present rules for calculations of resistance of trees in a Perturbed Markov Chain (PMC) in Section 2.3 of Chapter 2. The resistance of trees is a key enabler for analyzing the convergence of the proposed algorithms. Intuitively, the resistance of an edge of a tree is the cost of playing a suboptimal action. The proposed algorithms induce a perturbed Markov chain that can be described by a Transition Probability Function (TPF). Using the TPF of the algorithm the resistance can be calculated. However, the TPF can be composite and intricate that can make the calculation of resistance infeasible. Therefore, we develop rules to simplify the calculation of the resistance of trees for intricate and composite TPFs. These rules translate the resistance computation of composite functions into resistance computation of simple elementary functions. Thereby enabling easy analysis of the proposed learning algorithms. Particularly, we will see that these rules are essential in the proof of BLLA in noisy-potential games.

4. *Load Balancing in Heterogeneous Networks:*
   In Chapter 3, we apply the proposed learning algorithms in near-potential games for load balancing in heterogeneous networks that use CRE bias for user association and ABS for interference management. We first model the load balancing problem as a constrained minimization of an $\alpha-$fairness objective function with load and outage constraints. The $\alpha-$fairness objective function captures various network performances and fairnesses for different α. A rate-optimal policy is obtained for $\alpha = 0$, proportional fairness for $\alpha = 1$, minimum average delay for $\alpha = 2$, and min-max load policy as $\alpha \to \infty$. We provide a detailed proof for the min-max load policy for $\alpha \to \infty$. This proof extends the classical result derived in [1] by considering a non-convex $\alpha-$fairness function. Next, we model the above constrained minimization problem as a near-potential game in which base stations are the players and their

actions is a set of CRE biases and ABS ratios. We turned the objective function into the potential function of the near-potential game by designing the costs functions of base stations based on the neighborhood of the base stations. In presence of shadowing the neighborhood of base stations can be large and may theoretically include the whole network. We solve the problem distributively by constructing small approximate neighborhoods of the base stations. Doing so, we also propose a technique to iteratively construct the neighborhoods that allow the use of cost functions based on neighborhood. Then, the global minimum of the objective function is achieved by using the proposed LLA and BLLA in near-potential games. One important contribution is to derive practical conditions in terms of system parameters under which BLLA and LLA converge to the global minimum. By running extensive simulations, we show that the proposed algorithms converge to the global minimum.

5. *Annealing Learning Algorithm for LLA and BLLA:*
   In Chapter 4, we propose a new Annealing Learning Algorithm (ALA) to learn the temperature parameter τ of LLA and BLLA. ALA is developed by adapting successive reject algorithm [2]. In ALA, a set of values of τ of wide range and a time horizon is considered. The best τ value among a set of τ values is automatically obtained by ALA by evaluating the algorithms for the set of τ values. In the evolution of LLA, ALA checks the performance of different τ to successively reject the ones with the worst performance. At the end of the horizon, ALA selects the best performing τ. The process of successively rejecting the set of undesirable τ results into an annealing schedule for LLA and BLLA. We show that this annealing schedule is fast, performs better than other annealing schedules, and guarantees asymptotic convergence.

6. *Channel Assignment in D2D Networks:*
   In Chapter 5, we apply the proposed learning algorithms in noisy-potential games to channel assignment problem in D2D wireless networks. First, the CAP is modeled as an SOP, where the goal is to obtain the optimal average sum data rate of the network. Then, the SOP is translated into a noisy-potential game that takes into account the estimation noise of the data rate. We adapt the distributed BLLA in the noisy-potential game to achieve the optimal channel assignments. Extensive simulations show that the proposed BLLA achieves the maximum sum data rate of the network.

7. *Distributed Greedy Algorithm in Games with Submodular Objective Function:* In Chapter 6, we focus on a different setting other than potential games. We consider a setting where the objective function is submodular. We consider a general constrained submodular maximization problem. We model it as a finite game with players having limited or no information about their neighborhood. The information of neighborhood is represented as a Directed Acyclic Graph (DAG), which we called as information graph. We study the performance of a distributed greedy algorithm for the players with limited information. We first give lower and upper bounds on the worst-case performance of the greedy algorithm for a given information graph. The bounds give intuition on what properties of the underlying graph need to be improved in order for the performance to improve. This enables to determine the best information graph that gives the best performance that a system designer can achieve with a fixed number of agents and information edges. These results show that when information is costly, the best communication structures spread out the communication links among the agents, rather than clustering them among a small group.

1.2 Related Literature of Distributed Learning in Potential Games

With the increasing importance of distributed solutions, the learning in games is becoming a popular topic of research. Particularly, potential games are attractive as they enable distributed way of optimizing a potential function [3, 4]. Near-potential games widen the scope of potential games and share the similar convergence properties [5,6]. Noisy-potential games cover the problems with noisy utilities [7, 8].

A variety of distributed algorithms is available in the literature for potential games. In the following, we compare and contrast the algorithms for potential games that are most relevant to our work.

Best Response (BR) algorithm for exact potential games is the foremost algorithm whose dynamics are studied in the literature [3, 9–11]. In BR algorithm, the current player chooses the action that maximizes its utility given the strategies of other players. However, to maximize its utility the player has to know the utilities of all its actions. A BR is guaranteed to converge to an NE in exact potential games. However, the obtained NE may be suboptimal in general because the potential function may have multiple local optima [3, 9, 11].

To address the problem of getting trap into a local optimal a Log-linear learning algorithm (LLA) was introduced in [12, 13]. Since then the dynamics of LLA has been extensively studied in the literature [11–20]. In LLA, a player chooses its action with a probability that depends on its utility, usually larger probability for the actions with larger utility. The player chooses a suboptimal action sometimes with positive probability. This allows for escaping a local optimal and in the long run, the LLA converges to the global optimum of the potential function [20]. However, similar to as in BR algorithm, to calculate the probability of choosing an action using LLA the current player has to know the utilities of all its actions.

A variant of LLA called Binary LLA (BLLA) that requires the player to know only the utilities of two of its actions is proposed in the literature [17, 20]. In BLLA, a player chooses the action with a probability that depends on only two utilities of the player. In the long run BLLA is also guaranteed to converge to the global optimal of the potential function [17, 20].

Near-potential games are interesting as they share the similar properties of exact potential games. The dynamics of LLA for near-potential games have been studied in [5]. They prove that LLA converges to an approximate NE. In our work, we give proofs for both LLA and BLLA in near-potential games by extending to a more general framework by using a different proof technique. We use resistance trees of induced PMCs. The resistance expressions of transitions of LLA and BLLA are different. We use our proposed rules to calculate them. Our technique does not need stationary revision process. Instead, the revision process can be state and history dependent. Unlike in [5], our proof does not assume the underlying Markov chain to be reversible, which allows extending their result to the cases where players are chosen in a non-stationary and state-dependent way to revise their actions. As the underlying Markov chain is ergodic, each state has a positive probability to be chosen throughout the iterations of the algorithm.

Noisy-potential games were introduced to take into account the noisy utilities. BLLA for noisy-potential games has been originally proposed in [7]. However, the convergence of BLLA is proven for a fixed temperature. Whereas, in our work, we give proof for both fixed and decreasing temperature parameter of BLLA. Also, our proof takes into account the cases of bounded and unbounded noise. We give the sufficient number of samples that guarantee the convergence in both bounded and unbounded noise, unlike in [7].

Other closely related algorithms are log-linear trial and error learning [21], interactive trial and error learning [22], Adaptive learning [9]. However, these algorithms are for

general games, in which an NE may not even exist unlike in potential games. Therefore, we focus on learning algorithms in potential games.

The related literature on distributed greedy algorithm for submodular maximization is given in Chapter 6.

Chapter 2

Distributed Learning in Potential Games

In this chapter, we present distributed learning in potential games for distributively solving optimization problems. This applies to problems both deterministic and stochastic in nature. The goal is to distributively obtain the global optimal of the objective function rather than analyzing the local myopic behavior of the players of the game. For this purpose potential games provide a good framework. We first introduce different classes of potential games in Section 2.1. Then, we describe the proposed distributed algorithms for near-potential games and noisy-potential games and state the convergence results in Section 2.2. Next, we describe how the convergence of algorithms can be analyzed by using the theory of Perturbed Markov Chains (PMC) and resistance trees in Section 2.3. Also, in this section, we present the proposed rules for computing the resistance of trees of a PMC. Finally, we give detailed proofs of convergence of algorithms in Sections 2.4 and 2.5.

2.1 Potential Games

We consider a finite game $\Gamma = \left\{\mathscr{S}, \{X_i\}_{i\in\mathscr{S}}, \{U_i\}_{i\in\mathscr{S}}\right\}$, where $\mathscr{S}$ is the set of players, $X = X_1 \times X_2 \times \ldots \times X_{|\mathscr{S}|}$ is a strategy set or an action set, and $U_i : X \to \mathscr{R}$ is a utility or cost function. Different games such as are obtained with different designs of cost functions. A strategy profile of the game is denoted by $x = (x_i, x_{-i})$, where x_i is the strategy of player i and x_{-i} is the strategy profile all the players except player i. The players repeatedly play their actions to arrive at some equilibrium point if it exists. In an approximate or $\varepsilon-$NE no player can benefit more than ε by changing its strategy unilaterally. Formally, $\varepsilon-$NE is

defined as below.

Definition 1 *[$\varepsilon-$Nash Equilibrium:] A strategy profile $\left(x_i^*, x_{-i}^*\right) \in X$ is an ε-NE if*

$$U_i(x_i^*, x_{-i}^*) - U_i(x_i, x_{-i}^*) \leq \varepsilon, \quad \forall x_i \in X_i, \forall i \in \mathscr{S}. \tag{2.1}$$

If $\varepsilon = 0$ then it is a Pure NE (PNE). If utility functions are considered instead of cost functions of players then the above inequality sign changes from $\leq$ to $\geq$.

2.1.1 Exact Potential Games

Definition 2 *[Exact potential game [3]] $\Gamma = \left\{\mathscr{S}, \{X_i\}_{i\in\mathscr{S}}, \{U_i\}_{i\in\mathscr{S}}\right\}$ is an exact potential game if there is a function called potential function $\Phi : X \to \mathscr{R}$ such that $\forall i \in \mathscr{S}$, $\forall x_i, x_i' \in X_i$ and $\forall x_{-i} \in X_{-i}$,*

$$U_i(x_i, x_{-i}) - U_i(x_i', x_{-i}) = \Phi(x_i, x_{-i}) - \Phi(x_i', x_{-i}). \tag{2.2}$$

An exact potential game has at least one PNE and local optimisers of the potential function are PNEs [3]. An optimal PNE is an action profile x^* that corresponds to the global optimum of the potential function. Furthermore, there exist several distributed learning algorithms that converge to a PNE when the game has a potential.

A potential game is obtained by particular designs of the cost functions. There exist many cost function designs that lead to a potential game, see [23]. Two of the important and simple cost function designs are Identical Interest Utility (IIU) and marginal contribution cost [23]. Potential game can also be obtained by the cost based on the neighborhood of the player [24, 25].

1. *Identical interest utility:* In this design, the cost function $U_i(x)$ is completely aligned with the potential function, i.e.,

$$U_i(x) = \Phi(x), \quad \forall i \in \mathscr{S}. \tag{2.3}$$

However, this design requires the players to know the information about all the other players to compute their costs $U_i(x)$. This kind of cost function design leads to a centralised approach for the optimisation problem. However, a distributed approach can be obtained by the following cost functions.

2. *Marginal contribution cost:* The marginal contribution cost of a player is defined as

$$U_i(x_i, x_{-i}) = \Phi(x_i, x_{-i}) - \Phi\left(x_i^0, x_{-i}\right), \tag{2.4}$$

where x_i^0 is a null action of the player i.

3. *Cost based on neighborhood:* The case where the potential function is a sum of functions $\Phi(x) = \sum_i \varphi_i(x)$, where φ_i represents the term of player i then the utility of players can be designed as below:

$$U_i(x_i, x_{-i}) = \sum_{j \in N_i} \varphi_j(x_i, x_{-i}), \tag{2.5}$$

where N_i is a neighborhood of player i. The players are neighbors if they affect each others utility. The neighborhood definition depends on the problem considered. In the next chapter we will see the definition of neighborhood in load balancing problem among the base stations of a wireless network.

2.1.2 Near-Potential Games

Even though cost functions design exists for potential games it is difficult to obtain potential games in many problems due to the constraint on the availability of information to the players. For example, in case the perfect information about the neighborhood is not available to the player then a potential game cannot be obtained. In this case of limited information on the neighborhood of the players, an approximate or near potential games can be obtained that share similar properties as potential games.

Based on the notion developed in [5] we now define near potential game or ξ-potential game as below.

Definition 3 *[ξ-potential game] A game $\mathscr{G} = \left\{\mathscr{S}, \{X_i\}_{i \in \mathscr{S}}, \{U_i\}_{i \in \mathscr{S}}\right\}$ is an ξ-potential game if there is a potential function $\Phi : X \to \mathscr{R}$ such that $\forall i \in \mathscr{S}$, $\forall x_i, x_i' \in X_i$ and $\forall x_{-i} \in X_{-i}$,*

$$\left|U_i(x_i, x_{-i}) - U_i(x_i', x_{-i}) + \Phi(x_i', x_{-i}) - \Phi(x_i, x_{-i})\right| \leq \xi. \tag{2.6}$$

For $\xi = 0$, it is an exact potential game [3]. The ξ captures the maximum pairwise difference between an ξ-potential game and an exact potential game with the same potential function as in [5, Definition 2.2].

As said earlier an exact potential game has at least one PNE and local optimisers of the potential function are PNEs [3]. In the following lemma, we provide the relationship between the PNEs of a potential game and a near-potential game with the same potential.

Lemma 1 *Let* $\mathscr{G} = \left\{\mathscr{S}, \{X_i\}_{i\in\mathscr{S}}, \{U_i\}_{i\in\mathscr{S}}\right\}$ *and* $\mathscr{G}' = \left\{\mathscr{S}, \{X_i\}_{i\in\mathscr{S}}, \{U_i'\}_{i\in\mathscr{S}}\right\}$ *be an exact potential game and a* ξ*-potential game respectively, sharing a common potential function* Φ*. If* x^* *is a PNE for* $\mathscr{G}$ *then it is a* ξ*-NE for* $\mathscr{G}'$*.*

Proof: Since, x^* is a PNE for $\mathscr{G}$, then for all players i and x_i we have:

$$U_i(x^*) - U_i(x_i, x^*_{-i}) \leq 0, \tag{2.7}$$

$$\Phi(x^*) - \Phi(x_i, x^*_{-i}) \leq 0. \tag{2.8}$$

The above inequalities follows from Definition 1 and 2, respectively. Then for $\mathscr{G}'$, we have:

$$\begin{aligned} U_i'(x^*) - U_i'(x_i, x^*_{-i}) &\leq U_i'(x^*) - U_i'(x_i, x^*_{-i}) + \Phi(x_i, x^*_{-i}) - \Phi(a^*), && (2.9)\\ &\leq \left|U_i'(x^*) - U_i'(x_i, x^*_{-i}) + \Phi(x_i, x^*_{-i}) - \Phi(x^*)\right|, && (2.10)\\ &\leq \xi. && (2.11) \end{aligned}$$

First inequality is obtained by adding a positive term and the third inequality is by Definition 3. It follows from the last inequality that x^* is a ξ-NE of $\mathscr{G}'$. ■

2.1.3 Noisy Potential Games

In some stochastic optimization problems, the goal is to find the maximum of the average of the objective function that has noise component. The noise can occur due to the nature of the problem or due to the lack of perfect information. We introduce noisy potential games that can model the noise of a stochastic problem for solving them distributively. We define a noisy potential games below.

Definition 4 *[Noisy potential game] Let the expected utility of player i be denoted as* $U_i = \mathbb{E}\left[\hat{U}_i\right]$*. The game* $\hat{\mathscr{G}} := \left\{\mathscr{D}, \{X_i\}_{i\in\mathscr{D}}, \left\{\hat{U}_i\right\}_{i\in\mathscr{D}}\right\}$ *is a noisy potential game if the game* $\mathscr{G} := \left\{\mathscr{D}, \{X_i\}_{i\in\mathscr{D}}, \{U_i\}_{i\in\mathscr{D}}\right\}$ *is a potential game with potential function* Φ*.*

All the cost function designs that lead to a potential games also lead to a noisy potential game if the function capture the noise of the problem. Particularly, noisy potential games can be obtained by considering the following utility function which represents the marginal

contribution of the player to the global utility function:

$$\hat{U}_i(x_i, x_{-i}) = \hat{\Phi}(x_i, x_{-i}) - \hat{\Phi}\left(x_i^0, x_{-i}\right), \tag{2.12}$$

where $\hat{\Phi}$ is a noisy realization of an objective function Φ and x_i^0 is a null action of player i. Note that random utility $\hat{U}$ may have a large variance that leads a noisy potential game to have a large deviation from the exact potential game. To reduce the variance of the utility we may define a sample mean of utility function:

$$\hat{U}_i^N = \frac{1}{N}\sum_{k=1}^{N} \hat{U}_i. \tag{2.13}$$

We will see in the next section that the number of samples N must be designed carefully so as to preserve the convergence properties of potential games.

Lemma 2 *For any N, the game* $\hat{\mathscr{G}}^N := \left\{\mathscr{D}, \{X_i\}_{i\in\mathscr{D}}, \{\hat{U}_i^N\}_{i\in\mathscr{D}}\right\}$ *is a noisy potential game with potential function* $\Phi(x)$.

2.2 Distributed Learning in Potential Games

Recall that the potential function property enables finding a global optimum using distributed learning algorithms. In this section, we analzye distributed learning algorithms in the context of near-potential games and noisy-potential games. We consider minimization of potential function using near-potential games and maximization of potential function using noisy-potential game.

2.2.1 Distributed Learning Algorithms in Near-Potential Games

In this section, we propose distributed learning algorithms for near-potential game to minimize a potential function. First, we present the Best Response (BR) algorithm and Log-linear Learning Algorithm (LLA) for the complete information setting. Next, the Binary Log-linear Learning Algorithm (BLLA) for the partial information setting is described.

Best Response Algorithm

Best response algorithm is an asynchronous algorithm where at any given time only a single player updates its strategy. The current player computes its cost $U_i(x_i, x_{-i}(t-1))$

for all $x_i \in X_i$ given the strategies $x_{-i} \in X_{-i}$ of other players. Then, the player chooses a strategy $x_i \in X_i$ that minimizes its cost. In other words, BS i chooses a strategy from its best response set B_i:

$$B_i(x_{-i}) = arg\,min_{x_i} U_i(x_i, x_{-i}). \tag{2.14}$$

Note that the BR algorithm requires complete information, i.e., the information of all the set of strategies of the player. Moreover, BR algorithm is not guaranteed to converge to the optimal PNE even in exact potential game because the potential function may have multiple local minima [3]. For a ξ-potential game, a PNE may not even exist. Therefore, a BR algorithm cannot be used for a near-potential game. A generalized version of the BR algorithm is log-linear learning algorithm that can escape a local minimum and reach the global minimum. This algorithm is described next.

Log-linear Learning Algorithm

LLA is a classical asynchronous algorithm that guarantees the convergence to the global minimum of the potential function of exact potential game [20]. LLA has also been proven to converge to approximate optimal NE of a near-potential game [5]. We also prove this in a more generalized setting using a different technique. LLA algorithm is similar to BR but allows deviations from the best response with a small probability. The deviations or perturbations are controlled by a temperature parameter τ. As the temperature τ increases the perturbation increases resulting in high probability of choosing a suboptimal action. As the temperature τ goes to zero the perturbation decreases making LLA choose the best response actions with higher probability. LLA is summarized in Algorithm 1.

LLA also require the players to have the complete information about the costs of their action sets. For example, given the strategies of others, the player has to know the cost function value for all its strategies to compute (2.15). With this information, it selects a strategy to play according to a probability distribution. In general, acquiring this amount of information is not feasible. To overcome this difficulty in the next subsection we propose to use BLLA.

Binary Log-linear Learning Algorithm

BLLA is a variant of LLA that requires the information of only two actions of the current player. This information requirement we called as partial information. Unlike complete information where the player has to know the effect of choosing any other actions from its

Algorithm 1 Log-linear Learning Algorithm

1: **Initialisation:** Start with arbitrary action profile x.
2: Set parameter τ.
3: **While** $t \geq 1$ **do**
4: Randomly select a player i.
5: Compute cost $U_i(x_i, x_{-i}(t-1))$ for all $x_i \in X_i$.
6: Take action $x_i(t)$ from X_i with probability $p_i^{x_i}(t)$,

$$p_i^{x_i}(t) = \frac{\exp\left(-\frac{1}{\tau}U_i(x_i, x_{-i}(t-1))\right)}{\sum_{x_i' \in X_i} \exp\left(-\frac{1}{\tau}U_i\left(x_i', x_{-i}(t-1)\right)\right)}. \tag{2.15}$$

7: All the other players must repeat their previous actions, i.e., $x_{-i}(t) = x_{-i}(t-1)$.

action set. BLLA is also an asynchronous algorithm. In this algorithm, the current player updates its action in two steps. In the first step, the player tries an action from its action set to obtain its cost. In the second step, the player chooses from the two actions (present and trial actions) with some probability as summarized in Algorithm 7. BLLA also has a temperature parameter τ that controls the perturbation. We say that the algorithm *converges* to a state if the probability of the state is nonzero when temperature τ goes to zero. Or, say otherwise, that the state is *stochastically stable*, see, e.g., [20]. A common practice in the literature is to empirically find a suitable temperature parameter τ. Usually, the value of temperature τ depends on specific applications. A small value of temperature may works well in many applications. We will study the effect temperature τ by applying BLLA to load balancing application in wireless networks in Chapter 3. We will see in Chapter 4 a learning algorithm for a choosing the best τ.

Algorithm 2 Binary Log-linear Learning Algorithm

1: **Initialisation:** Start with arbitrary action profile x.
2: Set parameter τ.
3: **While** $t \geq 1$ **do**
4: Randomly select a player i and a trial action $\hat{x}_i \in X_i$ with uniform probability.
5: Player i plays action $x_i(t-1)$ to compute the cost $U_i(x(t-1))$.
6: Player i plays the trial action $\hat{x}_i$ to compute its cost $U_i(\hat{x}_i, x_{-i}(t-1))$.
7: Player i selects action $x_i(t) \in (x_i(t-1), \hat{x}_i)$ with probability

$$\left(1 + e^{\Delta_i/\tau}\right)^{-1}, \tag{2.16}$$

where $\Delta_i = U_i(\hat{x}_i, x_{-i}(t-1)) - U_i(x(t-1))$.
8: All the other players repeat their previous actions, i.e., $x_{-i}(t) = x_{-i}(t-1)$.

Let Φ^* and $\Phi^\dagger$ be the first minimum and second minimum values of a function Φ. Let $|X|$ be the cardinality of the action space.

Theorem 1 *For any ξ-potential game $\mathscr{G}$ with potential Φ and $\varepsilon > 0$ the stochastically stable states of LLA and BLLA corresponds to a set of ξ-NEs with potential less than $\Phi^* + 2\xi\,(|X|-1)$ if*

$$\xi < \frac{\varepsilon}{2\,(|X|-1)}. \tag{2.17}$$

Proof: See Section 2.4.1. ■

Corollary 1 *For any ξ-potential game $\mathscr{G}$ with potential Φ the stochastically stable states of LLA and BLLA corresponds to a set of PNEs whose potential is Φ^* if*

$$\xi < \frac{\Phi^\dagger - \Phi^*}{2(|X|-1)}. \tag{2.18}$$

Proof: See Section 2.4.2. ■

The proof of convergence of LLA and BLLA to optimal PNE for an exact potential game is given in [20]. Following a similar technique, we prove the convergence of LLA and BLLA to the global minimum of the potential function of a ξ-potential game $\mathscr{G}$. The convergence proof of LLA in near-potential games is given in [5]. We give proofs for both LLA and BLLA in near-potential games by extending to a more general framework by using a different proof technique. We use resistance trees of induced PMCs. The resistance expressions of transitions of LLA and BLLA are different. We use our proposed rules in Section 2.3.3 to calculate the resistance. Our technique does not need stationary revision process. Instead the revision process can be state and history dependent. Unlike in [5], our proof does not assume the underlying Markov chain to be reversible, which allows extending their result to the cases where players are chosen in a non-stationary and state-dependent way to revise their actions. As the underlying Markov chain is ergodic, each state has a positive probability to be chosen throughout the iterations of the algorithm.

2.2.2 Distributed Learning Algorithm in Noisy-Potential Games

We consider a noisy potential game $\hat{\mathscr{G}}^N := \left\{\mathscr{D}, \{X_i\}_{i\in\mathscr{D}}, \left\{\hat{U}_i^N\right\}_{i\in\mathscr{D}}\right\}$ given in Corollary 2 and a corresponding exact potential game $\mathscr{G} := \left\{\mathscr{D}, \{X_i\}_{i\in\mathscr{D}}, \{U_i\}_{i\in\mathscr{D}}\right\}$ with potential function $\Phi(x)$ and $U_i = \mathbb{E}\left[\hat{U}^N\right]$. We now study BLLA for distributed learning in noisy potential games. The key idea of BLLA in the presence of noise is to use multiple samples

of the noisy cost functions to reduce the variance of the noise. By reducing the variance of noise in noisy potential game we can use the properties of potential games. The details of BLLA in presence of noise are described in Algorithm 3.

Algorithm 3 BLLA for noisy potential games

1: **Initialisation:** Start with arbitrary action profile x.
2: **While** $t \geq 1$ **do**
3: Set parameter $\tau(t)$.
4: Randomly select a player i and a trial action $\hat{x}_i \in X_i$ with uniform probability.
5: Player i plays action $x_i(t-1)$ for N times and computes the average cost $\hat{U}_i^N(x(t-1))$.
6: Player i plays the trial action $\hat{x}_i$ for N times to compute its average cost $\hat{U}_i^N(\hat{x}_i, x_{-i}(t-1))$.
7: At time t player i selects action $x_i(t) \in (x_i(t-1), \hat{x}_i)$ with probability

$$\left(1+e^{\Delta_i^N/\tau}\right)^{-1}, \tag{2.19}$$

where $\Delta_i^N = \hat{U}_i^N(x(t-1)) - \hat{U}_i^N(\hat{x}_i, x_{-i}(t-1))$.
8: All the other players repeat their previous actions, i.e., $x_{-i}(t) = x_{-i}(t-1)$.

Since the convergence of BLLA is governed by a temperature parameter τ, we study the affect of fixed τ and varying $\tau(t)$ on the convergence. Note that in [7], the convergence of BLLA for a fixed temperature is proven. However, we give proof for both fixed and decreasing temperature parameter of BLLA. Also, the results of convergence of BLLA for both the cases of bounded and unbounded noise are presented below. As the parameter τ goes to zero, the stationary distribution concentrates on a few states. Intuitively, the states whose limit probability is strictly positive as τ goes to zero are stochastically stable as defined in Definition 8. It is known that for exact potential games the stochastically stable states of BLLA are the maximizers of the potential function [20]. We extend this result to noisy potential games in the following theorem.

Theorem 2 *For any noisy potential game $\hat{\mathcal{G}}^N$ with potential Φ the stochastically stable states of BLLA are the global maximizers of the potential Φ if one of the following holds.*

1. *The estimation noise is bounded in an interval of size ℓ and the number of estimation samples used are*

$$N \geq \left(\log\left(\frac{4}{\xi}\right) + \frac{2}{\tau}\right) \frac{\ell^2}{2(1-\xi)^2\tau^2}, \tag{2.20}$$

where $0 < \xi < 1$.

2. *The estimation noise is unbounded with finite mean and variance. Let $M(\theta)$ be a finite moment generating function of noise. Let $\theta^* = arg\,max_\theta\, \theta\,(1-\xi)\,\tau - \log\,(M(\theta))$. The number of samples used are*

$$N \geq \frac{\log\left(\frac{4}{\xi}\right) + \frac{2}{\tau}}{\log\left(\frac{e^{\theta^*(1-\xi)\tau}}{M(\theta^*)}\right)}. \tag{2.21}$$

Proof: See Section 2.5. ■

The following corollary follows immediately for the case when the noise is assumed to have a Gaussian distribution.

Corollary 2 *Let the noise has the standard normal probability distribution. Then the stochastically stable states of BLLA are the global maximizers of the potential function if*

$$N \geq \frac{2\log\left(\frac{4}{\xi}\right) + \frac{4}{\tau}}{\tau^2\,(1-\xi)^2}. \tag{2.22}$$

A small N is desired for practical implementations. We can thus choose the lowest N that satisfies Theorem 2. In Theorem 2, we have a convergence in probability for fixed parameter τ. In Theorem 3, we consider the case of decreasing parameter τ for which we obtain an almost sure convergence to optimal state as in simulated annealing with the same annealing schedule [26].

Theorem 3 *Consider BLLA in a noisy potential game with a decreasing parameter $\tau(t) = 1/\log(1+t)$, and the number of samples $N(\tau)$ is given by Theorem 2. Then, BLLA converges with probability 1 to the global maximizer of the potential function.*

Proof: See Section 2.5.2 ■

2.3 Perturbed Markov Chains and Resistance Trees

The dynamics of the proposed learning algorithms can be analyzed using the trees of a perturbed Markov chain. In this section, we first give the preliminaries of perturbed Markov chains and resistance trees. Then, we present the proposed rules for computing the resistance of trees. These rules will be used for the proofs of the proposed algorithms in the next sections.

2.3.1 Priliminaries

In this subsection, we present preliminaries of resistance trees of a PMC. More details can be found in [9, 20]. A perturbed Markov process is characterized by a set $\{P^\tau\}$ of transition matrices over a state space X indexed by a parameter τ. Wherein, $\tau \in (0, \tau_h]$ is a parameter that controls the perturbation and τ_h is constant. P^0_{ab} and P^τ_{ab} denote the transition probabilities from state a to b in the unperturbed and the perturbed Markov chains, respectively. The definition of resistance of transitions and the definition of a regular perturbed Markov process are given below [9].

Definition 5 (Resistance of transition) *A perturbed Markov process $\{P^\tau\}$ is regular if it satisfies the following conditions [9]:*

1. $\exists \tau_h : \forall \tau \in (0, \tau_h]$, P^τ *is aperiodic and irreducible,*
2. $\lim_{\tau \to 0} P^\tau_{ab}$ *exists and is equal to* P^0_{ab},
3. *for each strictly positive TPF* P^τ_{ab} *there exists a non-negative number* R_{ab} *called the resistance of transition such that*

$$0 < \lim_{\tau \to 0^+} e^{\frac{R_{ab}}{\tau}} P^\tau_{ab} < \infty. \tag{2.23}$$

Note that if $P^0_{ab} > 0$ then $R_{\text{ab}} = 0$.

Definition 6 (Tree) *A tree, T, rooted at a state a, is a set of $|X| - 1$ directed edges such that, from every other state a' in the state space, there is a unique directed path in the tree to a.*

The resistance of the directed edge $a \to b$ is the resistance of P^τ_{ab}, hence R_{ab}. The resistance of a rooted tree, T, is the sum of the resistances on its edges

$$R(T) = \sum_{a,b \in T} R_{\text{ab}}. \tag{2.24}$$

Let $\mathscr{T}(a)$ be defined as the set of trees rooted at the state a. The stochastic potential of the state a is defined as

$$\gamma(a) = \min_{T \in \mathscr{T}(a)} R(T). \tag{2.25}$$

We now define a minimum resistance tree and the stochastically stable states of a PMC [9].

Definition 7 (Minimum Resistance Tree) *A minimum resistance tree is a tree that has the minimum stochastic potential, that is, any tree T that satisfies*

$$R(T) = \min_{a \in X} \gamma(a). \tag{2.26}$$

Definition 8 (Stochastically Stable State) *Let $\{P^\tau\}$ be a regular perturbed Markov process, and for each $\tau > 0$, let μ_τ be the unique stationary distribution of P^τ. A state "a" is stochastically stable if*

$$\lim_{\tau \to 0} \mu_\tau(a) > 0. \tag{2.27}$$

The following theorem by [9, Lemma 1] gives the relationship between the stochastically stable states and a minimum resistance tree.

Theorem 4 *Let $\{P^\tau\}$ be a regular perturbed Markov process, and for each $\tau > 0$, let μ_τ be the unique stationary distribution of P^τ. Then $\lim_{\tau \to 0} \mu_\tau$ exists and the limiting distribution μ_0 is a stationary distribution of P^0. The stochastically stable states are the roots of minimum resistance trees.*

The stochastically stable states of a PMC can be found by finding the roots of minimum resistance tree. LLA and BLLA induces a regular Markov process over the action space of an exact potential game [20]. Also, the stochastically stable states of LLA and BLLA are the maximizers of the potential function of the exact potential game [20]. Therefore, calculating the resistance of trees in the PMC is essential to characterize the convergence of LLA and BLLA. In the following, we first illustrate the calculation of resistance of BLLA in an exact potential game and then we present some simple rules for easy calculation of resistance. These rules will later become essential for calculation of the resistance of BLLA in near-potential games and in noisy-potential games.

2.3.2 Resistance of BLLA in Exact Potential Games

We illustrate the computation of resistance of BLLA in the exact potential game $\Gamma = \left\{\mathscr{S}, \{X_i\}_{i \in \mathscr{S}}, \{U_i\}_{i \in \mathscr{S}}\right\}$. More details can be found in [20]. This illustration also serves as a benchmark for comparison of calculations of resistance using the proposed rules.

BLLA induces a regular Markov process over the action space X of Γ [7, 20]. Let denote P^τ as the transition matrix of the regular Markov process. Let $m_i(t)$ denote the probability of choosing player i to revise its action. Lets consider a uniform probability $\frac{1}{|X_i|}$ of choosing a trial action by player i. Consider a feasible transition $a \to b$, where

$a \coloneqq (x_i(t-1), x_{-i}(t-1))$ and $b \coloneqq (x_i(t), x_{-i}(t-1))$, whose transition probability is given in (2.16). The transition probability P_{ab}^{τ} of BLLA is

$$P_{ab}^{\tau} = m_i(t) \frac{1}{|X_i|} \left(1 + e^{\Delta_i/\tau}\right)^{-1}. \tag{2.28}$$

Recall that $\Delta_i = U_i(b) - U_i(a)$. Substituting this in above equation, we have

$$P_{ab}^{\tau} = m_i(t) \frac{1}{|X_i|} \frac{\exp\left(\frac{1}{\tau} U_i(b)\right)}{\exp\left(\frac{1}{\tau} U_i(a)\right) + \exp\left(\frac{1}{\tau} U_i(b)\right)}. \tag{2.29}$$

Let $V(a,b) = \max\{U_i(a), U_i(b)\}$. Multiplying the numerator and denominator of the above equation by $e^{-\frac{V(a,b)}{\tau}}$, we obtain

$$P_{ab}^{\tau} = m_i(t) \frac{1}{|X_i|} \frac{e^{\left(\frac{1}{\tau}(U_i(b) - V(a,b))\right)}}{e^{\left(\frac{1}{\tau}(U_i(a) - V(a,b))\right)} + e^{\left(\frac{1}{\tau}(U_i(b) - V(a,b))\right)}}. \tag{2.30}$$

After simplifying the above equation, we obtain the below limit as

$$\lim_{\tau \to 0^+} \frac{P_{ab}^{\tau}}{\exp\left(\frac{V(a,b) - U_i(b)}{\tau}\right)} = \frac{m_i(t)}{|X_i| \, |V(a,b)|}. \tag{2.31}$$

Since, the above limit is positive and finite the induced process is a regular Markov process and the resistance of BLLA in exact potential game Γ according to Definition 5 is

$$R_{ab} = V(a,b) - U_i(b). \tag{2.32}$$

Let denote $\Delta_i^+ = \max\{0, \Delta_i\}$. Then the resistance is

$$R_{ab} = \Delta_i^+. \tag{2.33}$$

2.3.3 Resistance Computation Rules

The resistance in Definition 5 can be computed in case the transition probability function can be factorised into simple function and in case the limit in (2.23) can be evaluated. However, transition functions can be composite and intricate that may not always be simplified. Moreover, the limit in (2.23) cannot always be feasible to evaluate. For example, when $P_{ab}^{\tau} = \tau$, the limit cannot be evaluated. To overcome these limitations of Definition 5 we first give a new generalized definition of resistance that allows us to develop easy rules

to compute the resistance of any positive function.

Let $o(.)$ and $\omega(.)$ denote little "o" order and little "ω" order, respectively, that are defined below.

Definition 9 *Let f and g be two functions of τ. Then $f(\tau) \in o(g(\tau))$ if*

$$\lim_{\tau\to\infty} \frac{f(\tau)}{g(\tau)} = 0. \tag{2.34}$$

And $f(\tau) \in \omega(g(\tau))$ if

$$\lim_{\tau\to\infty} \left|\frac{f(\tau)}{g(\tau)}\right| = \infty. \tag{2.35}$$

Definition 10 (Resistance of positive function) *The resistance of a strictly positive function $f(\tau)$ is $Res(f)$ if there exists a strictly positive function $g(\tau)$ such that $g \in o\left(e^{k/\tau}\right)$ and $g \in \omega\left(e^{-k/\tau}\right)$ for any $k > 0$; and*

$$\lim_{\tau\to 0} \frac{f(\tau)}{g(\tau)e^{-\frac{Res(f)}{\tau}}} = 1. \tag{2.36}$$

Remark Note that Definition 10 includes Definition 5, in which $g(\tau) = \kappa, 0 < \kappa < \infty$. Now, we can evaluate the resistance of $P_{ab}^{\tau} = \tau$, i.e., $\mathrm{Res}(\tau) = 0$.

Remark Note that (2.36) is equivalent to

$$f(\tau) = g(\tau)e^{-\frac{\mathrm{Res}(f)}{\tau}} + h(\tau), \tag{2.37}$$

where $h(\tau) \in o\left(g(\tau)e^{-\frac{\mathrm{Res}(f)}{\tau}}\right)$.

Remark We call $g(\tau)$ as a *sub-exponential function* if $g \in o\left(e^{k/\tau}\right)$ and $g \in \omega\left(e^{-k/\tau}\right)$ for any $k > 0$. Note that it is equivalent to $|\log g| \in o\left(\frac{1}{\tau}\right)$.

Lemma 3 *Consider any two sub-exponential functions $g_1(\tau)$ and $g_2(\tau)$. Consider two real numbers R_1 and R_2. If $R_1 < R_2$ then*

$$g_2(\tau)e^{-R_2/\tau} \in o\left(g_1(\tau)e^{-R_1/\tau}\right). \tag{2.38}$$

Proof: Let k be a real number. Then

$$\lim_{\tau\to 0} \frac{g_2(\tau)e^{-R_2/\tau}}{g_1(\tau)e^{-R_1/\tau}} = \lim_{\tau\to 0} \frac{g_2(\tau)}{e^{(R_2-k)/\tau}} \left[\frac{g_1(\tau)}{e^{-(R_1-k)/\tau}}\right]^{-1}. \tag{2.39}$$

Since g_1 and g_2 are sub-exponential the above limit goes to zero when we choose $R_1 < k < R_2$. This is because the first factor goes to zero as $R_2 - k > 0$. Also, the second factor goes to zero as $R_1 - k < 0$. ■

Lemma 4 *If $Res(f)$ exists then it is unique.*

Proof: Assume that function f have two different resistances $R_1 < R_2$. Then, there exist g_1, g_2, h_1, h_2 such that

$$f(\tau) = g_1(\tau)e^{-\frac{R_1}{\tau}} + h_1(\tau) = g_2(\tau)e^{-\frac{R_2}{\tau}} + h_2(\tau), \tag{2.40}$$

where $h_1(\tau) \in o\left(g_1(\tau)e^{-\frac{R_1}{\tau}}\right)$ and $h_2(\tau) \in o\left(g_2(\tau)e^{-\frac{R_2}{\tau}}\right)$. Using Lemma 3, we have $h_2 \in o\left(g_1(\tau)e^{-\frac{R_1}{\tau}}\right)$. Rearranging terms in (2.40), we have

$$1 + \frac{h_1(\tau)}{g_1(\tau)e^{-\frac{R_1}{\tau}}} = \frac{g_2(\tau)e^{-\frac{R_2}{\tau}}}{g_1(\tau)e^{-\frac{R_1}{\tau}}} + \frac{h_2(\tau)}{g_1(\tau)e^{-\frac{R_1}{\tau}}}. \tag{2.41}$$

Using Lemma 3 to evaluate the limit of the above equation as τ goes to zero, we arrive at contradiction that $1 = 0$.

■

The following proposition gives important rules for computing $\mathrm{Res}(f)$.

Proposition 5 *Let f, f_1 and f_2 be strictly positive functions. Let $Res(f_1)$ and $Res(f_2)$ exist. Let κ be a positive constant.*

I $f_1(\tau)$ *is sub-exponential if and only if* $Res(f_1) = 0$. *In particular* $Res(\kappa) = 0$.

II $Res(e^{-\kappa/\tau}) = \kappa$.

III $Res(f_1 + f_2) = \min\{Res(f_1), Res(f_2)\}$.

IV *If* $Res(f_1) < Res(f_2)$ *then* $Res(f_1 - f_2) = Res(f_1)$.

V $Res(f_1 f_2) = Res(f_1) + Res(f_2)$.

VI $Res(\frac{1}{f}) = -Res(f)$.

VII *If* $\forall\tau, f_1(\tau) \leq f_2(\tau)$ *then* $Res(f_2) \leq Res(f_1)$.

VIII *If* $\forall\tau, f_1(\tau) \leq f(\tau) \leq f_2(\tau)$ *and if* $Res(f_1) = Res(f_2)$ *then* $Res(f)$ *exists and* $Res(f) = Res(f_1)$.

Remark In Rule IV, if $\text{Res}(f_1) = \text{Res}(f_2)$ then we cannot compute $\text{Res}(f_1 - f_2)$ because in general the difference of sub-exponential functions may not be a sub-exponential function. For example, choose $f_1(\tau) = 1 + e^{-k/\tau}$ and $f_2(\tau) = 1$ with $k > 0$ then $\text{Res}(f_1) = \text{Res}(f_2) = 0$ but $\text{Res}(f_1 - f_2) = k$.

Remark For Rule VIII, in general if $f_1(\tau) \leq f(\tau) \leq f_2(\tau)$ and $\text{Res}(f_1) \neq \text{Res}(f_2)$ then $\text{Res}(f)$ may not exist. For example, for $f(\tau) = \lambda(\tau) f_1 + (1 - \lambda(\tau)) f_2$, $\lambda(\tau) = \frac{1}{2}\left(\cos\left(\frac{1}{\tau}\right) + 1\right)$ $\text{Res}(f)$ does not exist.

Proof: Proof of Rule I: Let $f(\tau)$ be a sub-exponential function. Choosing $g(\tau) = f(\tau)$ and substituting $\text{Res}(f) = 0$ in (2.36) we get $\lim_{\tau \to 0} \frac{f(\tau)}{f(\tau) e^{-\frac{\text{Res}(f)}{\tau}}} = 1$. Therefore, we have $\text{Res}(f) = 0$.

Assume $\text{Res}(f) = 0$. From (2.37), we have $f(\tau) = g(\tau) + h(\tau)$, which is a sub-exponential function.

Let $f(\tau) = \kappa$ and $g(\tau) = \kappa$ then $g(\tau) \in o\left(e^{\frac{\kappa}{\tau}}\right)$ and $g(\tau) \in \omega\left(e^{-\frac{\kappa}{\tau}}\right)$, $\kappa > 0$. Substituting these in (2.36) we have $\text{Res}(\kappa) = 0$.

Proof of Rule II: Substituting $f(\tau) = e^{-\kappa/\tau}$ and $g(\tau) = 1$ in (2.36) we get $\text{Res}(f) = \kappa$.

Proof of Rule III: Let $\text{Res}(f_1)$ and $\text{Res}(f_2)$ be the resistances of functions f_1 and f_2, respectively. Then, from (2.37) we have $f_1(\tau) = g_1(\tau) e^{-\frac{\text{Res}(f_1)}{\tau}} + h_1(\tau)$, $f_2(\tau) = g_2(\tau) e^{-\frac{\text{Res}(f_2)}{\tau}} + h_2(\tau)$, where $h_1(\tau) \in o\left(g_1(\tau) e^{-\frac{\text{Res}(f_1)}{\tau}}\right)$, $h_2(\tau) \in o\left(g_2(\tau) e^{-\frac{\text{Res}(f_2)}{\tau}}\right)$. The sum of two functions can be written as

$$f_1(\tau) + f_2(\tau) = g_1(\tau) e^{-\frac{\text{Res}(f_1)}{\tau}} \left(1 + \frac{h_1(\tau)}{g_1(\tau) e^{-\frac{\text{Res}(f_1)}{\tau}}} \frac{g_2(\tau) e^{-\frac{\text{Res}(f_2)}{\tau}}}{g_1(\tau) e^{-\frac{\text{Res}(f_1)}{\tau}}} + \frac{h_2(\tau)}{g_1(\tau) e^{-\frac{\text{Res}(f_1)}{\tau}}}\right). \tag{2.42}$$

Consider the case when $\text{Res}(f_1) < \text{Res}(f_2)$. Using Lemma 3 we have $h_2 \in o\left(g_1(\tau) e^{-\frac{\text{Res}(f_1)}{\tau}}\right)$. Therefore, $f_1(\tau) + f_2(\tau) = g_1(\tau) e^{-\frac{\text{Res}(f_1)}{\tau}} + h_3(\tau)$, where $h_3(\tau) \in o\left(g_1(\tau) e^{-\frac{\text{Res}(f_1)}{\tau}}\right)$. According to (2.37), we have $\text{Res}(f_1 + f_2) = \text{Res}(f_1)$.

The case of $\text{Res}(f_1) = \text{Res}(f_2)$ leads to the same result as shown below.

$$f_1(\tau) + f_2(\tau) = e^{-\frac{\text{Res}(f_1)}{\tau}} \left[g_1(\tau) + g_2(\tau)\right] + h_1(\tau) + h_2(\tau). \tag{2.43}$$

Note that sum of sub-exponential functions $g_1(\tau) + g_2(\tau)$ is a sub-exponential function. Observe that $h_1(\tau) + h_2(\tau) \in o\left(\left[g_1(\tau) + g_2(\tau)\right] e^{-\frac{\text{Res}(f_1)}{\tau}}\right)$. As in the previous case, according to (2.37) we have $\text{Res}(f_1 + f_2) = \text{Res}(f_1)$

Proof of Rule IV: Also, it can be shown similarly to the proof of rule III that if $\text{Res}(f_1) < \text{Res}(f_2)$ then $\text{Res}(f_1 - f_2) = \text{Res}(f_1)$.

Proof of Rule V:

$$\lim_{\tau\to 0} \frac{f_1(\tau)f_2(\tau)}{g_1(\tau)g_2(\tau)e^{-\frac{\text{Res}(f_1)+\text{Res}(f_1)}{\tau}}} = \lim_{\tau\to 0} \frac{f_1(\tau)}{g_1(\tau)e^{-\frac{\text{Res}(f_1)}{\tau}}} \lim_{\tau\to 0} \frac{f_2(\tau)}{g_2(\tau)e^{-\frac{\text{Res}(f_2)}{\tau}}} = 1. \tag{2.44}$$

Therefore, $\text{Res}(f_1 f_2) = \text{Res}(f_1) + \text{Res}(f_2)$.

Proof of Rule VI: Since $\text{Res}(f)$ exists, inverting both sides of (2.36), we have

$$\lim_{\tau\to 0} \frac{f(\tau)}{g(\tau)e^{-\frac{\text{Res}(f)}{\tau}}} = 1 = \lim_{\tau\to 0} \frac{\frac{1}{f(\tau)}}{\frac{1}{g(\tau)}e^{-\frac{-\text{Res}(f)}{\tau}}}. \tag{2.45}$$

Note that $\frac{1}{g(\tau)}$ is sub-exponential. Therefore, we have $\text{Res}(\frac{1}{f}) = -\text{Res}(f)$.

Proof of Rule VII: Assume to the contrary $\text{Res}(f_1) < \text{Res}(f_2)$. Using Lemma 3, we have $g_2(\tau)e^{-\text{Res}(f_2)/\tau} \in o\left(g_1(\tau)e^{-\text{Res}(f_1)/\tau}\right)$ and $h_2 \in o\left(g_1(\tau)e^{-\text{Res}(f_1)/\tau}\right)$.

$$f_1 \le f_2, \tag{2.46}$$

$$g_1(\tau)e^{-\text{Res}(f_1)/\tau} + h_1(\tau) \le g_2(\tau)e^{-\text{Res}(f_2)/\tau} + h_2(\tau), \tag{2.47}$$

$$1 + \frac{h_1(\tau)}{g_1(\tau)e^{-\text{Res}(f_1)/\tau}} \le \frac{g_2(\tau)e^{-\text{Res}(f_2)/\tau} + h_2(\tau)}{g_1(\tau)e^{-\text{Res}(f_1)/\tau}}. \tag{2.48}$$

As $\tau \to 0$, we arrive at a contradiction that $1 \le 0$. Therefore, $\text{Res}(f_1) \ge \text{Res}(f_2)$.

Proof of Rule VIII: We have $1 \le \frac{f(\tau)}{f_1(\tau)} \le \frac{f_2(\tau)}{f_1(\tau)}$ and $\text{Res}\left(\frac{f_2(\tau)}{f_1(\tau)}\right) = \text{Res}(f_2) - \text{Res}(f_1) = 0$. By Rule I $\frac{f_2(\tau)}{f_1(\tau)}$ is sub-exponential. This implies that $\frac{f(\tau)}{f_1(\tau)}$ is also sub-exponential. Therefore, there exists $g_{01}(\tau)$ such that

$$1 = \lim_{\tau\to 0} \frac{\frac{f(\tau)}{f_1(\tau)}}{g_{01}(\tau)} = \lim_{\tau\to 0} \frac{f(\tau)}{g_{01}(\tau)g_1(\tau)e^{-\frac{\text{Res}(f_1)}{\tau}}} \lim_{\tau\to 0} \frac{g_1(\tau)e^{-\frac{\text{Res}(f_1)}{\tau}}}{f_1(\tau)}, \tag{2.49}$$

$$= \lim_{\tau\to 0} \frac{f(\tau)}{g_{01}(\tau)g_1(\tau)e^{-\frac{\text{Res}(f_1)}{\tau}}}, \tag{2.50}$$

where the product $g_{01}(\tau)g_1(\tau)$ is also a sub-exponential function. Therefore, $\text{Res}(f)$ exists and $\text{Res}(f) = \text{Res}(f_1) = \text{Res}(f_2)$.

■

These rules help simplify the compution of resistance of transition of a PMC. These rules reduce the computation of resistance of composite and intricate transition probability functions into the computation of resistance of simple functions. These rules are simple and yet are powerful. The strength of these rules is illustrated by using them to calculate efficiently the resistance of transition of the proposed learning algorithms in the following sections. Thus, these rules provide an efficient tool that can be used to characterize the stochastically stable states of learning algorithms whose dynamics is modelled by a PMC.

2.4 Convergence of LLA and BLLA in Near-Potential Games

2.4.1 Proof of Theorem 1

Proof: The proof is given for BLLA only. Whereas, for LLA it can be done following the similar approach. Note that for the ξ-potential game $\mathscr{G}$ both the LLA and BLLA induces a regular perturbed Markov process over the action space $X = X_1 \times X_2 \times \ldots X_{|\mathscr{S}|}$. We first compute the resistance of BLLA in ξ-potential game $\mathscr{G}$ using the proposed rules. Then, we proof the theorem using the obtained resistance.

Let $m_i(t)$ denote the probability of choosing player i to revise its action. Lets consider a uniform probability $\frac{1}{|X_i|}$ of choosing a trial action by player i. Consider a feasible transition $a \to b$, where $a \coloneqq (x_i(t-1), x_{-i}(t-1))$ and $b \coloneqq (x_i(t), x_{-i}(t-1))$, whose transition probability is given in (2.16). The transition probability P^{τ}_{ab} of BLLA is

$$P^{\tau}_{ab} = m_i(t)\frac{1}{|X_i|}\frac{\exp\left(-\frac{1}{\tau}U_i(b)\right)}{\exp\left(-\frac{1}{\tau}U_i(a)\right)+\exp\left(-\frac{1}{\tau}U_i(b)\right)}. \tag{2.51}$$

Using Rule V of Proposition 5, we have the resistance of transition as below:

$$\mathrm{Res}\left(P^{\tau}_{ab}\right) = \mathrm{Res}\left(m_i(t)\frac{1}{|X_i|}\right) + \mathrm{Res}\left(\frac{\exp\left(-\frac{1}{\tau}U_i(b)\right)}{\exp\left(-\frac{1}{\tau}U_i(a)\right)+\exp\left(-\frac{1}{\tau}U_i(b)\right)}\right). \tag{2.52}$$

The resistance $\text{Res}\left(m_i(t)\frac{1}{|X_i|}\right) = 0$ due to Rule I. Then, using Rules III and VI of Proposition 5, we have

$$\begin{aligned}\text{Res}\left(P_{ab}^{\tau}\right) = \text{Res}\left(\exp\left(-\frac{1}{\tau}U_i\left(b\right)\right)\right) \\ - \min\left\{\text{Res}\left(\exp\left(-\frac{1}{\tau}U_i(a)\right)\right), \text{Res}\left(\exp\left(-\frac{1}{\tau}U_i\left(b\right)\right)\right)\right\}. \quad (2.53)\end{aligned}$$

Using Rule II of Proposition 5, we have

$$\text{Res}\left(P_{ab}^{\tau}\right) = U_i\left(b\right) - \min\left\{U_i\left(a\right), U_i\left(b\right)\right\}. \quad (2.54)$$

This can be rewritten simply as

$$\text{Res}\left(P_{ab}^{\tau}\right) = U_i\left(b\right) - U_i\left(a\right). \quad (2.55)$$

Using the Definition 3, we obtain

$$\text{Res}\left(P_{ab}^{\tau}\right) \leq \Phi\left(b\right) - \Phi\left(a\right) + \xi. \quad (2.56)$$

We now proof the theorem using the above expression of resistance. Let a^* be a state with the global minimum potential Φ^*. Action profile a^* is an optimal ξ-NE of ξ-potential game and an optimal PNE of the corresponding exact potential game according to Lemma 1. Recall Theorem 1 and suppose by contradiction that a minimum resistance tree, $\mathscr{T}$, is rooted at an action profile a whose potential is more than $\Phi^* + 2\xi\left(|X| - 1\right)$. This means the action profile a is such that

$$\Phi(a) - \Phi^* \geq 2\xi\left(|X| - 1\right). \quad (2.57)$$

Since $\mathscr{T}$ is a rooted tree, there exists a path $\mathscr{P}$ from a^* to a of the form:

$$\mathscr{P} = \left\{a^* \to a^1 \to \ldots \to a^m \to a\right\}. \quad (2.58)$$

Consider the reverse path $\mathscr{P}^r = \left\{a \to a^m \to \ldots \to a^1 \to a^*\right\}$. The difference in the resistance of the paths can be calculated using (2.56) and is bounded by:

$$R\left(\mathscr{P}^r\right) - R\left(\mathscr{P}\right) \leq \Phi^* - \Phi(a) + \sum_{a,b\in\mathscr{P}} \xi + \sum_{c,d\in\mathscr{P}^r} \xi. \quad (2.59)$$

Construct a new tree $\mathscr{T}^1$ rooted at a^* by adding the edges of the reverse path $\mathscr{P}^r$ to $\mathscr{T}$ and removing the redundant edges $\mathscr{P}$. The new tree will have the following resistance:

$$R(\mathscr{T}^1) = R(\mathscr{T}) + R(\mathscr{P}^r) - R(\mathscr{P}). \tag{2.60}$$

Using the bound from (2.59) in the above equation, we obtain

$$R(\mathscr{T}^1) \leq R(\mathscr{T}) + \Phi^* - \Phi(a) + \sum_{a,b \in \mathscr{P}^r} \xi + \sum_{c,d \in \mathscr{P}} \xi. \tag{2.61}$$

Upper bounding with $(|X|-1)$ the total number of edges in the tree $\mathscr{T}^1$, we obtain

$$R(\mathscr{T}^1) < R(\mathscr{T}) + \Phi^* - \Phi(a) + 2\,(|X|-1)\,\xi. \tag{2.62}$$

Substituting $\varepsilon = \Phi(a) - \Phi^*$, we have

$$R(\mathscr{T}^1) \leq R(\mathscr{T}) - \varepsilon + 2\,(|X|-1)\,\xi. \tag{2.63}$$

We have $R(\mathscr{T}^1) < R(\mathscr{T})$ if $\xi < \frac{\varepsilon}{2(|X|-1)}$. Therefore, we can construct the tree $\mathscr{T}^1$ rooted at a^* with strictly less resistance than the tree $\mathscr{T}$ leading to a contradiction that $\mathscr{T}$ is minimal resistance tree. ■

2.4.2 Proof of Corollary 1

Proof: By substituting $2\xi\,(|X|-1) = \Phi^\dagger - \Phi^*$ in Theorem 1, we obtain that the algorithm converges to a set of states with potential strictly less than $\Phi^* + (\Phi^\dagger - \Phi^*) = \Phi^\dagger$. Hence, the only possible states are those with potential value Φ^*. Let a^* be one such state, and assume that it is not a PNE of ξ-potential game. Then there exists a player i and an action x_i, such that: $U_i(a^*) - U_i(a_i, a^*_{-i}) > 0$. Since $\Phi(a_i, a^*_{-i}) - \Phi(a^*) \geq \Phi^\dagger - \Phi^* = \varepsilon$ it follows $\left|U_i(a^*) - U_i(a_i, a^*_{-i}) + \Phi(a_i, a^*_{-i}) - \Phi(a^*)\right| > \varepsilon$ which is a contradiction with the game $\mathscr{G}$ being a ξ-potential game with $\xi < \frac{\varepsilon}{2|X|}$. ■

2.5 Convergence of BLLA in Noisy-Potential Games

2.5.1 Proof of Convergence of BLLA with Fixed τ

For this proof[1], we show that for a carefully chosen number of samples N the resistance of BLLA in noisy potential game with noisy utilities $\hat{U}_i^N$ is equal to that of in the corresponding exact potential game with deterministic utilities U_i. This results in the transition probability of BLLA in noisy potential games equal to the transition probability of BLLA in the exact potential game. Since BLLA converges in exact potential game it also converges in noisy potential game. This kind of proof idea based on resistance can also be found in [7, 9].

In the following, we show that the resistance of BLLA for the noisy potential game $\hat{\mathscr{G}}^N$ with estimated utilities $\hat{U}_i^N$ is same as in (2.33). For this, we need the following lemma.

Lemma 5 *Let denote* $\Delta_i^N = \hat{U}_i^N(a) - \hat{U}_i^N(b)$, $\Delta_i = U_i(a) - U_i(b)$,

$$p_i^N = \mathbb{E}\left[\left(1 + e^{\Delta_i^N/\tau}\right)^{-1}\right], \tag{2.64}$$

$$p_i = \left(1 + e^{\Delta_i/\tau}\right)^{-1}, \tag{2.65}$$

and for some $\delta > 0$ *consider the event* $A^\delta = \left\{\left|\Delta_i^N - \Delta_i\right| < \delta\right\}$. *Then*

$$\left|p_i^N - p_i\right| \leq \delta\tau^{-1}p_i + 2Pr\left(\bar{A}^\delta\right). \tag{2.66}$$

Proof: A proof similar to the below can be found in [7, Proposition 2]. Here, for the sake of completeness we give detailed proof. Notice that the probability of transition of BLLA from action profile a to b in noisy potential game $\hat{\mathscr{G}}^N$ is $p_i^N = \Pr^N(a \to b)$ given in (2.64) and in deterministic potential game is $p_i = \Pr(a \to b)$ given in (2.65). Using the law of total probability, we can write

$$p_i^N = \Pr^N\left(a \to b|A^\delta\right)\Pr\left(A^\delta\right) + \Pr^N\left(a \to b|\bar{A}^\delta\right)\Pr\left(\bar{A}^\delta\right) \tag{2.67}$$

$$\left|p_i^N - p_i\right| \leq \left|\Pr^N\left(a \to b|A^\delta\right) - p_i\right|\Pr\left(A^\delta\right) + \left|\Pr^N\left(a \to b|\bar{A}^\delta\right) - p_i\right|\Pr\left(\bar{A}^\delta\right). \tag{2.68}$$

It can be shown that the absolute value of the derivative of p_i with respect to Δ_i is

[1] In all the proofs in the following, the considered utilities are normalized by the maximum potential $\phi_{\max}$.

$\tau^{-1} p_i (1-p_i) \leq \tau^{-1} p_i$. Therefore, we have

$$\left| \Pr^N \left(a \to b | A^\delta \right) - p_i \right| \leq \delta \tau^{-1} p_i. \tag{2.69}$$

Also, we bound $\left| \Pr^N \left(a \to b | \bar{A}^\delta \right) - p_i \right| \leq 2$. Substituting, this and (2.69) in (2.68) we have (2.66). ■

Proof: [Proof for bounded noise case] Let denote the noise $Z_i = \hat{U}_i(a) - U_i(a) - \left(\hat{U}_i(b) - U_i(b)\right)$. Using Hoefding inequality for bounded independent random variables, we have

$$\Pr\left(\bar{A}^\delta\right) = \Pr\left(\frac{1}{N}\sum_{i=1}^{N} |Z_i| > \delta\right) \leq 2\exp\left(-2N\frac{\delta^2}{\ell^2}\right). \tag{2.70}$$

Substituting (2.70) in Lemma 5, we have

$$p_i\left(1-\frac{\delta}{\tau}\right) - 4e^{-2N\frac{\delta^2}{l^2}} \leq p_i^N \leq p_i\left(1+\frac{\delta}{\tau}\right) + 4e^{-2N\frac{\delta^2}{\ell^2}}. \tag{2.71}$$

Substituting the number of samples N from (2.20) and choosing $\delta = (1-\xi)\,\tau$ in above, we have

$$\xi\left(p_i - e^{-\frac{2}{\tau}}\right) \leq p_i^N \leq (2-\xi)\,p_i + \xi e^{-\frac{2}{\tau}}. \tag{2.72}$$

As before, the transition probability of BLLA is $P_{ab}^\tau = \frac{m_i(t)}{|X_i|} p_i^N$. Using this in the above inequality, we obtain

$$\frac{m_i(t)}{|X_i|}\xi\left(p_i - e^{-\frac{2}{\tau}}\right) \leq P_{ab}^\tau \leq \frac{m_i(t)}{|X_i|}(2-\xi)\,p_i + \frac{m_i(t)}{|X_i|}\xi e^{-\frac{2}{\tau}}. \tag{2.73}$$

In the following, we calculate the resistance of lower and upper bound of the above P_{ab}^τ using Proposition 5. Note that $\mathrm{Res}(p_i) = \Delta_i^+$ as given in (2.33), $\mathrm{Res}(e^{-\frac{2}{\tau}}) = 2$, and $\Delta_i \leq 2$. The resistance of lower bound of P_{ab}^τ is

$$\mathrm{Res}\left(\frac{m_i(t)}{|X_i|}\xi\left(p_i - e^{-\frac{2}{\tau}}\right)\right) = \mathrm{Res}\left(\frac{m_i(t)}{|X_i|}\xi\right) + \mathrm{Res}\left(\left(p_i - e^{-\frac{2}{\tau}}\right)\right), \tag{2.74}$$

$$= \min\left\{\mathrm{Res}\left(p_i\right), \mathrm{Res}\left(e^{-\frac{2}{\tau}}\right)\right\}, \tag{2.75}$$

$$= \mathrm{Res}(p_i). \tag{2.76}$$

Similarly, the resistance of upper bound of P_{ab}^{τ} is

$$\mathrm{Res}\frac{m_i(t)}{|X_i|}\left((2-\xi)\,p_i+\frac{m_i(t)}{|X_i|}\xi e^{-\frac{2}{\tau}}\right)=\min\left\{\mathrm{Res}\left(\frac{m_i(t)}{|X_i|}(2-\xi)\,p_i\right),\mathrm{Res}\left(\frac{m_i(t)}{|X_i|}\xi e^{-\frac{2}{\tau}}\right)\right\}, \tag{2.77}$$

$$=\min\left\{\mathrm{Res}\,(p_i),\mathrm{Res}\left(e^{-\frac{2}{\tau}}\right)\right\}, \tag{2.78}$$

$$=\mathrm{Res}(p_i). \tag{2.79}$$

Since both the bounds have the same resistance, by Rule VIII the resistance of P_{ab}^{τ} exists and is equal to $\mathrm{Res}(p_i)$. Therefore, the resistance of transitions of BLLA with bounded noise is same as in the case of without noise (2.33). ■

Proof: [Proof for unbounded noise case] In this case, we use Chernoff bound to calculate $\Pr\left(\bar{A}^{\delta}\right)$ because of the unbounded noise as below. Let denote the noise Z_i with moment generating function $M(\theta)$.

$$\Pr\left(\bar{A}^{\delta}\right)=\Pr\left(\frac{1}{N}\sum_{i=1}^{N}|Z_i|>\delta\right)=2\Pr\left(\frac{1}{N}\sum_{i=1}^{N}Z_i>\delta\right), \tag{2.80}$$

$$\leq 2\exp\left(-N\log\left(\frac{e^{\theta^*\delta}}{M(\theta^*)}\right)\right), \tag{2.81}$$

where, (2.80) is obtained by assuming symmetric probability distribution of noise. However, for non-symmetric distribution a more complex expression can be obtained. Also, we used the Chernoff bound for independent and identically distributed random variables to obtain the equation (2.81).

Substituting (2.81), $\delta=(1-\xi)\,\tau$, and the number of samples N from (2.21) in Lemma 5, we have

$$\xi\,p_i-4\xi e^{-\frac{2}{\tau}}\leq p_i^N\leq(2-\xi)\,p_i+4\xi e^{-\frac{2}{\tau}}. \tag{2.82}$$

Following the same steps as before, we get that the resistance of transitions of BLLA with unbounded noise is same as in the case of without noise (2.33).

■

2.5.2 Proof of Convergence of BLLA with Decreasing $\tau(t)$

We follow the proof approach as in [27] by Anily and Federgruen. The Theorem 1 in [27] cannot be used directly since our TPF does not belong to the class of asymptotically

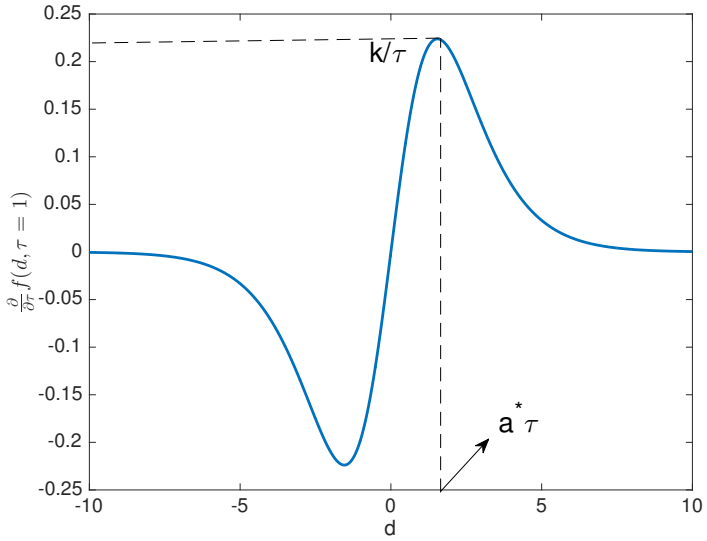


Figure 2.1: This figure shows the function $\frac{\partial f(d,\tau)}{\partial \tau} = \frac{d}{\tau^2} \frac{1}{2+\exp(d/\tau)+\exp(-d/\tau)}$ of Lemma 6.

monotone function (CAM) or rationally closed class of bounded variation (RCBV) functions (see [27, Def. 3 and 4]). We give the proof in the case of bounded noise. The proof for unbounded noise can be done similarly.

Proof: [Proof of Theorem 3] We check that the assumptions of Theorem 1 in [27] are satisfied for the proof of Theorem 3. In Lemma 8, we prove that BLLA generates a weakly ergodic non-homogeneous Markov chain. In Lemma 9, we show that the stationary distribution $\pi(\tau)$ of the homogeneous Markov chain is a bounded variation function of τ. ■

We now give the Lemmas required for the above proof in the following. To simplify the notations we omit to specify the index of player i and particular transition when not needed. For a given parameter τ, we set $N(\tau)$ as in (2.20), and we consider $p(\tau) = p^{N(\tau)}$. Recall that $p(\tau) = \mathbb{E}\left[f(\Delta^N, \tau)\right]$ with $f(d,\tau) = \left(1+\exp\left(\frac{d}{\tau}\right)\right)^{-1}$. We denote $\delta = \mathbb{E}\left[\Delta^N\right]$.

Lemma 6 *For a given τ, function $\dfrac{\partial f(d,\tau)}{\partial \tau}$ is odd, has the sign of d, is bounded in absolute value by k/τ for some $k > 0$, and the maximum is attained (for positive value) at the point $a^*\tau$, where $a^* > 0$ and it is independent of τ.*

Proof: We have the function

$$\frac{\partial f(d,\tau)}{\partial \tau} = \frac{d}{\tau^2} \frac{1}{2+\exp(d/\tau)+\exp(-d/\tau)}, \tag{2.83}$$

which is shown in Fig. 2.1. This is an odd function in d that has the sign of d. Hence, we

just consider the case $d > 0$. Then

$$\frac{\partial^2 f(d,\tau)}{\partial d \partial \tau} = \frac{1}{\tau^2(2+Y+Y^{-1})^2}\left[1 - \frac{d}{\tau}\frac{Y-Y^{-1}}{2+Y+Y^{-1}}\right],$$

with $Y = \exp(d/\tau)$. This is first positive and then negative when d is positive. The maximum is reached when

$$\frac{d}{\tau}\frac{Y-Y^{-1}}{2+Y+Y^{-1}} = 1. \tag{2.84}$$

We claim that the maximum in d is attained for $d^* = a^*\tau$, with $a^* > 0$ a constant. Indeed, consider $d = a\tau$ with $a > 0$ in (6.2), which gives

$$2 + \exp(a)(1-a) + \exp(-a)(1+a) = 0.$$

Consider the function $g(a) = 2+\exp(a)(1-a)+\exp(-a)(1+a)$. We have $g(0) = 4$, and g tends to $-\infty$ when a goes to ∞. Furthermore, the derivative is $-a(\exp(a)+\exp(-a))$ which is strictly negative, hence there is a unique solution a^* to the equation (6.2). Replacing d by $d^* = a^*\tau$ in (2.83) yields:

$$\frac{\partial f(d^*,\tau)}{\partial \tau} = \frac{a^*}{\tau}\frac{1}{2+\exp(a^*)+\exp(-a^*)}.$$

Hence the result follows with $k = \dfrac{a^*}{2+\exp(a^*)+\exp(-a^*)}$. ■

Lemma 7 *If $\delta > 0$ (resp. $\delta < 0$), then $p(\tau)$ is increasing (resp. decreasing) in the vicinity of $\tau = 0$. Furthermore, $|p'(\tau)|$ has resistance $|\delta|$.*

Proof: We consider $\delta > 0$. The case $\delta < 0$ is similar.

We will show that the derivative $p'(\tau)$ is positive in the vicinity of 0. Previous lemma shows that $\dfrac{\partial f(d,\tau)}{\partial \tau} \leq k/\tau$. Since the constant function k/τ is integrable w.r.t. to the distribution of Δ^N, then

$$p'(\tau) = \mathbb{E}\left[\frac{\partial f(\Delta^N,\tau)}{\partial \tau}\right]. \tag{2.85}$$

By previous lemma, the point reaching the maximum of $\dfrac{\partial f(d,\tau)}{\partial \tau}$ is $a^*\tau$, then it goes to zero when τ goes to zero, and the function is then decreasing. Hence, for any $\varepsilon < \delta$, there is τ small enough such that the minimum (resp. maximum) of the derivative on the interval

$[\delta-\varepsilon,\delta+\varepsilon]$ is attained at $\delta+\varepsilon$ (resp. $\delta-\varepsilon$). Consider the event

$$A^{\varepsilon}=\left\{\left|\Delta^{N}-\delta\right|<\varepsilon\right\}. \tag{2.86}$$

In the following, we proof the Lemma by bound the $p'(\tau)$.

$$\begin{aligned}
p'(\tau) &= \mathbb{E}\left[\frac{\partial f(\Delta^N,\tau)}{\partial\tau}\right], && (2.87)\\
&= \mathbb{E}\left[\frac{\partial f(\Delta^N,\tau)}{\partial\tau}|\bar{A}^{\varepsilon}\right]\mathbb{P}[\bar{A}^{\varepsilon}]+\mathbb{E}\left[\frac{\partial f(\Delta^N,\tau)}{\partial\tau}|A^{\varepsilon}\right]\mathbb{P}[A^{\varepsilon}], && (2.88)\\
&\geq -\frac{k}{\tau}\mathbb{P}[\bar{A}^{\varepsilon}]+\frac{\partial f(\delta+\varepsilon,\tau)}{\partial\tau}\mathbb{P}[A^{\varepsilon}], && (2.89)\\
&\geq -\frac{k\xi}{\tau}\exp(-\frac{2}{\tau})+0.5\frac{\partial f(\delta+(1-\xi)\tau,\tau)}{\partial\tau}. && (2.90)
\end{aligned}$$

In the above (2.89) is obtained by using Lemma 6 and (2.90) is obtained by choosing $\varepsilon=(1-\xi)\tau$ and N is given by Theorem 2. The resistance of the first term $\text{Res}\left(-\frac{k\xi}{\tau}\exp(-\frac{2}{\tau})\right)=2$ and resistance of second term $\text{Res}\left(0.5\frac{\partial f(\delta+(1-\xi)\tau,\tau)}{\partial\tau}\right)=\delta<2$. Therefore, the first term is negligible compared to the second term. Hence the lower bound is positive for small enough τ and the resistance of lower bound on $p'(\tau)$ is δ. Hence the derivative is lower bounded by a positive function and then is positive.

The upper bound is obtained with the following inequality:

$$\begin{aligned}
p'(\tau) &= \mathbb{E}\left[\frac{\partial f(\Delta^N,\tau)}{\partial\tau}|\bar{A}^{\varepsilon}\right]\mathbb{P}[\bar{A}^{\varepsilon}]+\mathbb{E}\left[\frac{\partial f(\Delta^N,\tau)}{\partial\tau}|A^{\varepsilon}\right]\mathbb{P}[A^{\varepsilon}],\\
&\leq \frac{k}{\tau}\mathbb{P}[\bar{A}^{\varepsilon}]+\frac{\partial f(\delta-\varepsilon,\tau)}{\partial\tau}\mathbb{P}[A^{\varepsilon}],\\
&\leq \frac{k\xi}{\tau}\exp(-\frac{2}{\tau})+0.5\frac{\partial f(\delta-(1-\xi)\tau,\tau)}{\partial\tau}.
\end{aligned}$$

This upper bound has the resistance δ. Therefore, from Rule VIII we have that the resistance of $p'(\tau)$ is δ. ■

Lemma 8 *The non-homogeneous Markov chain generated by the BLLA algorithm with decreasing parameter $\tau(t)=\frac{1}{\log(1+t)}$ is weakly ergodic.*

Proof: The conditions of validity of Theorem 2 in [28] are checked by Lemma 7, Equation (2.82) and the classical choice of decreasing parameter τ. More details about weak ergodicity can also be found in [29]. ■

If a real valued function f is defined on the interval $[a,b]$, f is differentiable and its derivative f' is Riemann integrable then its total variation $V_a^b(f)$ is

$$V_a^b(f) = \int_a^b \left|f'(x)\right| dx. \tag{2.91}$$

f is a bounded variation function if its total variation is finite i.e., $V_a^b(f) < \infty$. In particular, if the derivative f' is bounded then $V_a^b(f) < \infty$ and f is a bounded variation function.

Let $\pi(\tau)$ be the stationary distribution of the homogeneous Markov chain for a given τ.

Lemma 9 *$\pi(\tau)$ has a bounded derivative.*

Proof: By the Markov chain tree theorem [30] for every state $c \in X$, we have $\pi_c(\tau) = \frac{u(c)}{\sum_{d\in X} u(d)}$ where

$$u_c(\tau) = \sum_{T\in\mathscr{T}_c} \prod_{e\in T} p_e(\tau), \tag{2.92}$$

$p_e(\tau)$ (2.64) is the transition probability to a state e and $\mathscr{T}_c$ is the set of trees rooted at the state c. Then

$$\begin{aligned}
\left|\pi_c'(\tau)\right| &= \left|\left(\frac{u_c}{\sum_d u_d}\right)'\right|, \\
&\leq \frac{|u_c'|(\sum_d u_d)}{(\sum_d u_d)^2} + \frac{u_c\left|\sum_d u_d'\right|}{(\sum_d u_d)^2}, \\
&\leq \frac{|u_c'|}{\sum_d u_d} + \frac{|\sum_d u_d'|}{\sum_d u_d}, \\
&\leq \frac{|u_c'|}{\sum_d u_d} + \sum_d \frac{|u_d'|}{\sum_{d'} u_{d'}}.
\end{aligned}$$

Hence it suffices to show that $\dfrac{|u_c'|}{\sum_d u_d}$ is bounded for all states c.

Let $\mathrm{Res}(T)$ denotes the total resistance of a tree T and $R_{\min}$ denotes the resistance of the minimal resistance tree. By using Proposition 5, we obtain

$$\mathrm{Res}\left(\sum_d u_d(\tau)\right) = \mathrm{Res}\left(\sum_d \sum_{T\in\mathscr{T}_d} \prod_{e\in T} p_e(\tau)\right), \tag{2.93}$$

$$= \min_{d\in X} \min_{T\in\mathscr{T}_d} \mathrm{Res}\left(\prod_{e\in T} p_e(\tau)\right), \tag{2.94}$$

$$= \min_{d\in X, T\in\mathscr{T}_d} \mathrm{Res}(T), \tag{2.95}$$

$$= R_{\min}. \tag{2.96}$$

The derivative of transition probability $u_c'(\tau)$ is obtained using (2.92) as

$$u_c'(\tau) = \sum_{T \in \mathscr{T}_c} \sum_{e \in T} p_e'(\tau) \prod_{d \in T/e} p_d(\tau). \tag{2.97}$$

The resistance of $u_c'(\tau)$ is

$$\mathrm{Res}\left(u_c'\right) = \min_{T \in \mathscr{T}_c} \min_{e \in T} \left[\mathrm{Res}\left(p_e'\right) + \sum_{d \in T/e} \mathrm{Res}\left(p_d(\tau)\right) \right]. \tag{2.98}$$

By Lemma 7, $\mathrm{Res}\left(p_e'\right)$ is $|\delta|$. If $\delta > 0$ then $\mathrm{Res}\left(p_e\right) = \delta$, otherwise $\mathrm{Res}\left(p_e\right) = 0$, which corresponds to the best response transition. Hence, $\mathrm{Res}\left(p_e'\right) \geq \mathrm{Res}\left(p_e\right)$ and $\mathrm{Res}\left(p_e'\right)$ is strictly greater than $\mathrm{Res}\left(p_e\right)$ if $\delta < 0$. In Lemma 10, we prove that the minimal resistance tree must contain a transition with null resistance (which corresponds to the best response). Hence, the state c at which $R_{\min}$ is reached contains at least one transition with $\delta \leq 0$. Therefore, $\mathrm{Res}\left(u_c'\right) > R_{\min}$. Using Proposition 5, we have $\mathrm{Res}\left(\frac{|u_c'|}{\sum_d u_d}\right) = \mathrm{Res}\left(u_c'|\right) - R_{\min} > 0$. Hence, $\frac{|u_c'|}{\sum_d u_d} \to 0$ as τ goes to zero for all states c. This finally shows that the derivative $|\pi_c'(\tau)|$ is bounded. ■

Lemma 10 *A minimum resistance tree must contain a transition with zero resistance.*

Proof: Assume that a minimum resistance tree $T_{\min}$ have all the transitions with non-zero resistance. Let the root of this tree be a state s and let there be a transition from another state s' to s. Let $R_{s' \to s}$ be a non-zero resistance of this transition. Note that the resistance of reverse transition $R_{s \to s'} = 0$ because it corresponds to the best response transition. Construct a new tree T rooted at state s' by adding the transition $s \to s'$ and removing the transition $s' \to s$. The resistance of the tree T is

$$R_T = R_{T_{\min}} - R_{s' \to s} + R_{s \to s'}, \tag{2.99}$$

$$< R_{T_{\min}}. \tag{2.100}$$

We arrive at a contradiction. Therefore, a minimum resistance tree must contain a transition with null resistance. ■

Chapter 3

Load Balancing in Heterogeneous Wireless Networks using Distributed Learning in Near-Potential Games

In this chapter, we present a novel approach for distributed load balancing in heterogeneous networks that use cell range expansion (CRE) for user association and almost blank subframe (ABS) for interference management. First, we formulate the problem as a minimisation of an $\alpha-$fairness objective function with load and outage constraints. Depending on α, different objectives in terms of network performance or fairness can be achieved. Next, we model the interactions among the base stations for load balancing as a near-potential game, in which the potential function is the $\alpha-$fairness function. The optimal pure Nash equilibrium (PNE) of the game, which correspond to the global minimum of $\alpha-$fairness function, is found using distributed learning algorithms developed in the previous chapter. We adapt log-linear and binary log-linear learning algorithms for complete and partial information settings, respectively. We provide sufficient conditions under which the learning algorithms converge to the optimal PNE. By running extensive simulations, we show that the proposed algorithms converge within few hundreds of iterations. The convergence speed in the case of partial information setting is comparable to that of the complete information setting. Finally, we show that outage can be controlled and a better load balancing can be achieved by introducing ABS. pathloss

3.1 Introduction

Due to the ever increasing demand for improved quality of service in terms of higher data rates and improved coverage, the conventional cellular networks are becoming heterogeneous. Heterogeneous networks consist of macro base stations (BSs) and small BSs that transmit with high and low power, respectively. Conventional user association rule is such that the users select a BS that provides the highest received power. This may however result in an imbalance between BSs loads because the macro BSs transmit at higher power and thus associates with more users. This creates overload situation at the macro BSs and at the same time under-utilized resources at the small BSs. Therefore, a natural problem that arises is how to associate users to BSs such that the network resources are utilised efficiently and the load is shared among the BSs.

Load balancing has been extensively studied in the literature using various approaches. An overview can be found in [31, 32]. These can be broadly classified as centralized, e.g. in [33–36], and decentralised optimisation approaches, see e.g. [37–41]. However, centralized solutions are computationally extensive, require huge information exchange overhead, and are thus not scalable. To overcome these limitations, decentralized approaches have been proposed. Load balancing is modeled as a convex optimization problem in [37] and a distributed algorithm, which is a fixed point iteration, is proposed to solve it. It is a user-centric approach, in which users decide to which BS they associate based on load information broadcast by BSs. User-centric game theoretical and learning approaches are also proposed, e.g. using congestion games [38, 40], evolutionary games [39] or distributed Q-learning [41].

Some network-centric approaches include power control and cell range expansion (CRE). In power control, more users can be offloaded to small BSs by increasing their transmit power because user association is based on maximum received power of user [42]. Since small BSs have tight power constraint, power control may not be feasible for efficient load balancing. In this chapter, we focus on an alternative network-centric approach, in which BSs take decisions and users follow a predefined association rule called CRE. According to the CRE technique, users associate with a BS that provides the maximum biased received power. A CRE bias is broadcast by every BS and is typically higher for small BSs than for macro BSs. This results in an increase of the small cell coverage and thereby of the number of users associated with them. CRE technique has the drawback of increasing outage probability at the cell edge [43], it is therefore often deployed in conjunction with almost blank subframes (ABS) at the macro BS [44]. During these

subframes that represent a fixed ratio of the radio frame, macro BSs drastically lower their transmit power, so that small BSs cell edge users can experience less interference when scheduled during these periods.

The challenge we intend to tackle here is to jointly determine the optimal CRE bias values and ABS ratios for a required optimal performance of the network. Several papers try to achieve a similar goal. In the first set of papers, performance evaluation and optimization are performed using simulations or experiments [45–47]. They give interesting insights but provide optimal values only for some specific scenarios. Centralized approaches provide upper bounds on performance but fail to address the scalability issue, see e.g. [48–52]. In [49] for example, authors formulate an optimization problem aiming at maximizing the Jain's fairness index between station loads.

Another set of papers is focused on distributed algorithms [41, 48, 53–56]. Early papers [48,53,54] propose heuristics without any goal of achieving some kind of optimality. In [56], authors formulate an integer programming problem, relaxed into a convex problem, and they propose a distributed algorithm based on Lagrangian dual decomposition. ABS and related tradeoffs are however ignored. In [55], two independent algorithms are presented for optimal CRE bias and ABS ratio, respectively. Optimal parameters are obtained for a given number of active users and the outage constraint is ignored. In a recent paper [57], Liu et al. model the system as a potential game and use the best response algorithm to reach a Nash Equilibrium (NE) that is also a local optimizer of a proportional fairness objective function. In this paper, as in others (e.g. [52, 55, 56]), a static *full buffer* traffic model is assumed. This implies that proposed distributed algorithms should converge faster than the change of traffic conditions. This is an assumption that does not seem realistic for practical implementations. On the contrary, we assume a dynamic traffic model and we update association parameters after the traffic has reached a stationary state. This is also the approach of [37,47], which respectively focus on user-centric load balancing (without CRE or ABS) and a centralized approach. Very few analytical works (with the exception of [41]) consider outage probability as a possible constraint when using CRE. Nowhere in the above literature, the effect of shadowing is studied in conjunction with CRE and ABS.

In this chapter, we propose a general framework for determining jointly the optimal CRE biases and the optimal ABS ratios for different performance requirements of the network. We address this problem by considering an $\alpha-$fairness objective function that captures various aspects of the network performance and fairness for different α values. A similar function is used in [37] but in a different context with the main difference that our

problem is not convex. We model our system using the notion of near-potential game [5], a framework needed to take into account shadowing in the radio propagation model. We solve the non-convex optimization problem using distributed learning algorithms, which converge in probability to the best NE of the game even in absence of complete information (in contrast with [57]).

3.1.1 Contributions

We summarize our main contributions of this chapter below.

- *Novel Approach:* We present a novel approach for load balancing in heterogeneous networks that use CRE bias for user association and ABS for interference management. Our approach is to distributively solve a constrained minimization of an $\alpha-$fairness objective function with load and outage constraints using the proposed distributed learning algorithms in near-potential games. The $\alpha-$fairness objective function captures various network performances and fairnesses for different α. A rate-optimal policy is obtained for $\alpha = 0$, proportional fairness for $\alpha = 1$, average delay for $\alpha = 2$, and min-max load policy as $\alpha \to \infty$. We provide a detailed proof for the min-max load policy for $\alpha \to \infty$. This proof extends the classical result derived in [1] by considering a non-convex $\alpha-$fairness function.

- *Near-potential game framework:* We model the above constrained minimization problem as a near-potential game in which base stations are the player and their actions is a set of CRE biases and ABS ratios. We turned the objective function into the potential function of the near-potential game by designing the costs functions of BSs based on the neighborhood of the BSs. In presence of shadowing the neighborhood of BSs can be large and may include the whole network. We solve the problem distributively by constructing small approximate neighborhoods of the BSs. Doing so, we also propose a technique to iteratively construct the neighborhoods that allow the use of cost functions based on neighborhood.

- *Learning algorithms:* By adapting log-linear learning algorithms LLA and BLLA to outage and load constraints we achieve the global minimum of the objective function. We consider two different settings: complete and partial information. In the former setting, we adapt classical LLA, whereas, in the latter setting, we adapt BLLA. The convergence of LLA and BLLA in near-potential games to an ξ-NE, whose potential

is close to the global minimum of the objective function is proved in Theorem 1. Next, we provide sufficient conditions on the parameter that controls the size of the neighborhood, for convergence to an ξ-NE with potential close to the global optimum of the objective function (Theorem 8). We also provide stronger conditions for the convergence to the optimal PNE (Corollary 4). To the best of our knowledge, these conditions are not present in the literature. Finally, we propose a practical step by step construction of the neighborhood with guaranteed convergence based on mobile users measurements (Section 3.4.4).

- *Numerical results:* By running extensive simulations, we show that the proposed algorithms converge within few hundreds of iterations to the global minimum. The convergence speed of the BLLA is comparable to that of the LLA, meaning that partial information is sufficient in practical implementations. We show that under pathloss scenario load balancing can be solved using exact potential games since the neighborhoods are small. We show that when shadow fading is also considered along with pathloss then the neigbhorhood becomes large and may include the whole network. In this scenario, near-potential games becomes apt for use because a reduced approximate neighborhood can be used. We show that outage can be controlled without ABS but at the price of high outage probability undermining the interest of using CRE technique. The introduction of ABS allows for low outage together with better load balancing.

The rest of the chapter is organized as follows. In Section 3.2, the system model is described and the problem is formulated. In Section 3.3, a near-potential game framework solution is presented. Distributed algorithms are described in Section 3.4. In Section 3.5, our approach is validated using extensive simulations. Finally, the conclusions are given in Section 3.6. All the proves are relegated to Appendix 3.7.

3.2 System Model

3.2.1 Network Model

We consider the downlink[1] of a cellular network (typically a LTE-Advanced network) consisting of a set $\mathscr{B}_e$ of macro BSs (typically eNodes-B) and a set $\mathscr{B}_s$ of small BSs in a

[1] Downlink is usually considered as the dominant link in terms of traffic. However, optimal user association on the downlink may not be optimal for the uplink [58].

two dimensional region $\mathscr{L}$. The set of all stations is denoted $\mathscr{S} \triangleq \mathscr{B}_e \cup \mathscr{B}_s$. The transmit powers of macro and small BSs are denoted as P_{macro} and P_{small}, respectively. There are special subframes called ABS, during which a macro BS transmits with reduced power P_{ABS}. The proportion of ABS subframes is denoted $\theta_i \in [0;1]$ for BS i. Let $\bar{\boldsymbol{\theta}} = \left[\theta_1, \theta_2, \ldots, \theta_{|\mathscr{S}|}\right]$ be the ABS ratio vector. We assume that small BSs do not implement ABS technique, i.e., $\theta_i = 0$ for $i \in \mathscr{B}_s$. Every small BS i maintains a parameter $c_i \in [1; c_{\max}]$ called CRE bias. The CRE bias vector is denoted $\bar{\boldsymbol{c}} = [c_1, c_2, \ldots, c_{|\mathscr{S}|}]$. The CRE biases for macro BSs are fixed to unity, i.e., $c_k = 1, \forall k \in \mathscr{B}_e$. This leads to no bias in the received power from a macro BS.

3.2.2 Channel Model

The received power at location x from BS i is $P_i g_i(x)$, where P_i is the transmit power and $g_i(x)$ is the channel gain, which captures the effect of pathloss and shadowing. The effect of small-scale fading is not considered because the time for user association procedure is assumed to be much larger than the channel coherence time [37]. We consider a scenario where the locations of the BSs and of the users during their download are fixed. Therefore, the shadow fading component is a constant multiplicative factor. Formally, the channel gain model considered is [59]:

$$g_i(x) = \min\left\{1, K\left|x - x_i\right|^{-\eta} e^{\beta y_i(x)}\right\}, \tag{3.1}$$

where $K = \left(\frac{\lambda_w}{4\pi d_0}\right)^2$, λ_w is the wavelength, d_0 is the reference distance, x_i is the location of the BS i, $\eta \geq 2$ is the pathloss exponent, and $e^{\beta y_i(x)}$ is the shadowing component where $\beta = \frac{\log 10}{10}$ and $y_i(x)$ is a realisation of Gaussian random process of zero mean and covariance function $C_{y_i}(\Delta x)$ [60]:

$$C_{y_i}(\Delta x) = \sigma_{sh}^2 e^{-\frac{\Delta x}{D_c}}, \tag{3.2}$$

where σ_{sh}^2 is the variance, Δx is the displacement, and D_c is the decorrelation distance [59]. A constant cross correlation between the $y_i(x)$ and $y_j(x)$ is considered as in [61].

Let M be the number subframes in a given radio frame. For every allowed value of θ, we assume that there is a fixed ABS pattern $\Upsilon(\theta)$, i.e., a set of subframes during which a BS transmits at lower power. Notice that $\Upsilon(0) = \varnothing$. Then the SINR $\gamma_i^f(x, \bar{\boldsymbol{\theta}})$ of a user at

location x in a subframe f is given as:

$$\gamma_i^f(x,\bar{\boldsymbol{\theta}}) = \frac{P_i^f(\theta_i)g_i(x)}{\sum_{j\in\mathscr{S}} P_j^f(\theta_j)g_j(x)+N_0}, \tag{3.3}$$

where

$$P_i^f(\theta_i) = \begin{cases} P_{\text{ABS}} & \text{if} \quad f \in \Upsilon(\theta_i),\ i \in \mathscr{B}_e, \\ P_{\text{macro}} & \text{if} \quad f \notin \Upsilon(\theta_i),\ i \in \mathscr{B}_e, \\ P_{\text{small}} & \text{otherwise} \end{cases} \tag{3.4}$$

and $N_0 = -174 + 10\log W$ is thermal noise power in dBm and W is bandwidth in Hz.

3.2.3 CRE User Association Rule

A user association rule based on CRE and maximum transmit power is commonly used in the heterogeneous networks [31, 43–45, 62–68]. According to this rule, a user located at x is associated to the BS i that provides the highest biased received power. The set of locations $\mathscr{D}_i(\bar{\boldsymbol{c}})$ associated to BS i is defined as:

$$\mathscr{D}_i(\bar{\boldsymbol{c}}) = \left\{x | \forall j \in \mathscr{S}, P_i g_i(x) c_i \geq P_j g_j(x) c_j\right\}, \tag{3.5}$$

where $P_i = P_{macro}$ if $i \in \mathscr{B}_e$ and P_{small} otherwise.

Physical Data Rate

The physical data rate received by a user at x in a subframe f when it is served by BS i is denoted $\tilde{\nu}_i^f(x,\bar{\boldsymbol{\theta}})$. The user average data rate over a radio frame is $\nu_i(x,\bar{\boldsymbol{\theta}}) = \frac{1}{M}\sum_{f=1}^{M}\tilde{\nu}_i^f(x,\bar{\boldsymbol{\theta}})$. This should be understood as the throughput achievable by the user alone in its cell. The function $\tilde{\nu}_i^f(x,\bar{\boldsymbol{\theta}})$ is a non-negative and non-decreasing function of the SINR $\gamma_i^f(x,\bar{\boldsymbol{\theta}})$. For SINR below minimum threshold $\gamma_{\min}$ the user is not served and $\tilde{\nu}_i^f(x,\bar{\boldsymbol{\theta}}) = 0$.

Traffic Model

Users are assumed to arrive in the system according to a spatial random process, download a file of random size and leave the system when the download is over. This is referred to as elastic traffic. All users are scheduled in all subframes[2] [55]. At location x, the arrival rate

[2] Another scheduling policy where only those users are served that are in the extended region can also be included in our model [55]. However, it adds more complexity without affecting the conclusions of this work.

is denoted $\lambda(x)$ [arrivals/s/m^2] and the average file size is $1/\mu(x)$ [bits]. Following [37], we model every BS i as a M/G/1/PS queue of load:

$$\rho_i(\bar{\boldsymbol{c}},\bar{\boldsymbol{\theta}}) = \int_{\mathscr{D}_i(\bar{\boldsymbol{c}})} \frac{\lambda(x)}{\mu(x)\nu_i(x,\bar{\boldsymbol{\theta}})} 1_{\left\{\max_f \gamma_i^f(x,\bar{\boldsymbol{\theta}})\geq\gamma_{\min}\right\}} dx. \tag{3.6}$$

BS i is stable if and only if $0 \leq \rho_i < 1$. In this work, only stable network configurations are considered.

Assumption 1 *[Time-scale separation] The process of updating the CRE bias and ABS ratio is supposed to be long with respect to the traffic variations. The M/G/1/PS queues describing the BSs traffic are thus supposed to have reached their stationary regime before any new change of these parameters.*

Outage Probability is defined as the fraction of users that are not served. Recall that a user is not served if its SINR is below minimum threshold $\gamma_{\min}$. Formally, the outage probability O_i observed by BS i is given by:

$$O_i(\bar{\boldsymbol{c}},\bar{\boldsymbol{\theta}}) = \frac{\int_{\mathscr{D}_i(\bar{\boldsymbol{c}})} \lambda(x) 1_{\left\{\max_f \gamma_i^f(x,\bar{\boldsymbol{\theta}})<\gamma_{\min}\right\}} dx}{\int_{\mathscr{D}_i(\bar{\boldsymbol{c}})} \lambda(x) dx}. \tag{3.7}$$

In this definition, as soon as there is at least one subframe during which the SINR is above the threshold, the user is supposed to be served.

3.2.4 Problem Formulation and Objective Function

Following [37], we intend to minimise an $\alpha-$fairness function $\phi_\alpha\left(\bar{\boldsymbol{c}},\bar{\boldsymbol{\theta}}\right)$ over a feasible set $\mathscr{F}$, which are defined as:

$$\phi_\alpha(\bar{\boldsymbol{c}},\bar{\boldsymbol{\theta}}) = \begin{cases} \sum_{i\in\mathscr{S}} \frac{(1-\rho_i(\bar{\boldsymbol{c}},\bar{\boldsymbol{\theta}}))^{1-\alpha}}{\alpha-1}, & \alpha \geq 0, \alpha \neq 1, \\ -\sum_{i\in\mathscr{S}} \log\left(1-\rho_i(\bar{\boldsymbol{c}},\bar{\boldsymbol{\theta}})\right), & \alpha = 1, \end{cases} \tag{3.8}$$

$$\mathscr{F} = \left\{\left\{\bar{\boldsymbol{c}},\bar{\boldsymbol{\theta}}\right\} | \forall i \in \mathscr{S}, \rho_i(\bar{\boldsymbol{c}},\bar{\boldsymbol{\theta}}) < 1, O_i(\bar{\boldsymbol{c}},\bar{\boldsymbol{\theta}}) < \bar{O}_i,\right\}, \tag{3.9}$$

where $\bar{O}_i$ is the maximum outage probability for BS i. The function $\phi_\alpha(\bar{\boldsymbol{c}},\bar{\boldsymbol{\theta}})$ is in general non-convex and even if it is convex the set $\mathscr{F}$ is non-convex because $\bar{\boldsymbol{c}}$ takes discrete values. The function $\phi_\alpha(\bar{\boldsymbol{c}},\bar{\boldsymbol{\theta}})$ captures various aspects of fairness and performance for the network depending on the choice of α.

($\alpha = 0$) **Min-sum-load policy:** Minimising $\phi_0(\bar{\boldsymbol{c}}, \bar{\boldsymbol{\theta}})$ minimizes the sum of BSs loads in general. In the particular case, where $\bar{\boldsymbol{\theta}} = 0$, it results in a rate-optimal policy (see Appendix 3.7.1). ($\alpha = 1$) **Proportional fair policy:** Minimising $\phi_1(\bar{\boldsymbol{c}}, \bar{\boldsymbol{\theta}})$ is equivalent to achieving proportional fairness between BSs [1]. ($\alpha = 2$) **Delay-optimal policy:** It can be shown that minimising $\phi_2(\bar{\boldsymbol{c}}, \bar{\boldsymbol{\theta}})$ is equivalent to minimising the average throughput of the network. The average throughput of a stable M/G/1/PS queue is the product of arrival rate and average delay. In our system model, the arrival rate is independent of CRE bias and ABS ratio. Therefore, minimising $\phi_2(\bar{\boldsymbol{c}}, \bar{\boldsymbol{\theta}})$ is equivalent to minimising the average delay of the network. For more detailed discussion refer to [37]. ($\alpha \to \infty$) **Minmax policy:** As $\alpha \to \infty$ the minimiser of $\phi_\alpha(\bar{\boldsymbol{c}}, \bar{\boldsymbol{\theta}})$ tends to the min-max load vector. It is a standard result with convex objective function [1, 37, 69]. We now prove this result for our non-convex objective function in Theorem 6.

Definition 11 *[Min-max load vector [69]] Let all the vectors in $\mathscr{F}$ be sorted in increasing order. A vector $\rho \in \mathscr{F}$ is min-max if ρ is lexicographically not greater than any vector in $\mathscr{F}$. The vector ρ is* lexicographically lower *than y, denoted $\rho \prec y$, if the first non-zero component of $\rho - y$ is negative. We say that ρ is not greater than y, denoted by $\rho \preceq y$, if $\rho \prec y$ or $\rho = y$.*

Let $r_i(\bar{\boldsymbol{c}}) = 1 - \rho_i(\bar{\boldsymbol{c}}, \bar{\boldsymbol{\theta}}), \forall i \in \mathscr{S}$. Let $\mathscr{X} = \left\{ r \in \mathscr{R}^{|\mathscr{S}|} | \exists \bar{\boldsymbol{c}} : \rho(\bar{\boldsymbol{c}}) \in \mathscr{F}, r(\bar{\boldsymbol{c}}) = r \right\}$. Load vector ρ^* is a min-max if and only if r^* is max-min vector.

Theorem 6 *Let $r_\alpha \in \underset{r \in \mathscr{X}}{argmax} \sum_{i \in \mathscr{S}} \frac{r_i^{1-\alpha}}{1-\alpha}$. Then, any accumulation vector of the trajectory $\{r_\alpha\}_{\alpha > 1}$ is max-min in $\mathscr{X}$.*

Proof: See Appendix 3.7.2. ■

In Fig. 3.1, we show an example of $\mathscr{F}$ set obtained with 2 small BSs having different transmit powers located in a two-dimensional region. It is clear from the figure that even if the CRE set were continuous, $\mathscr{F}$ would not be convex. We also show the optimal loads obtained for different α values. All the optimal load points are located on the Pareto frontier. The point for $\alpha \geq 200$ in Fig. 3.1 is the min-max load point because a point of equal coordinates on the Pareto frontier is the min-max point.

3.3 Near-Potential Game Framework

In this section, we model the interactions of BSs for load balancing as a near-potential game. We first model the load balancing problem as a *user association game*, Then, we show that

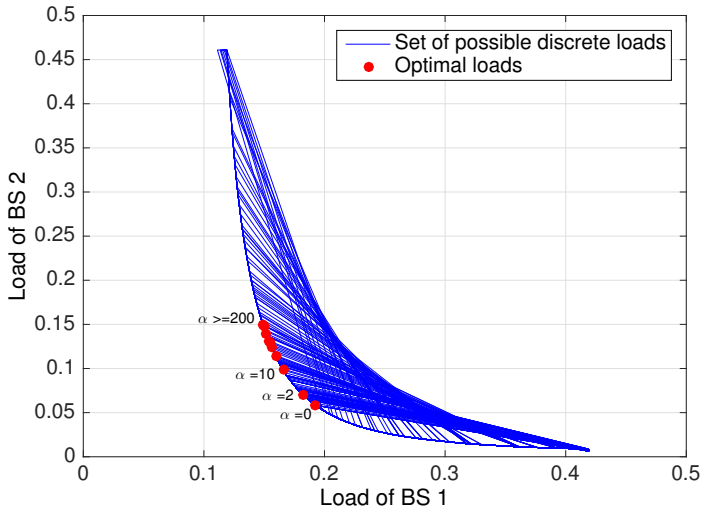


Figure 3.1: [Feasible set $\mathscr{F}$ for 2 small BSs]: We see that the optimal load points are located on the Pareto frontier of the feasible set. For $\alpha \geq 200$, the loads of the BSs are equal representing a load equalizing policy.

a near-potential game model is naturally suitable for the user association game due to the constraint on maintaining a small neighbourhood size so that a distributed algorithm can be designed. Otherwise, without the constraint on the size of the neighborhood, the user association game can be modelled as an exact potential game. However, with the larger neighbourhood, the distributed nature is compromised, which is not desirable.

3.3.1 User Association Game

A *user association game* is a finite game with the BSs as players and their strategies are allowable CRE bias and ABS ratio values. In this subsection we design the cost functions of BSs to model the user association game as a near-potential game. We seek that the objective function (5.3) is turned into a potential function of the near-potential game. We interchangeably say near-potential game or ξ-potential game, where the parameter ξ represents a tradeoff between the quality of the solution and the distributed nature of the algorithm. The issue is in designing the cost functions of the BSs to obtain a ξ-potential game that we address below.

We consider a simple cost structure for BS i, which takes into account only the effects of its neighbours. A similar approach is used to obtain an exact potential game in [24, 25].

The cost functions of the individual BSs are defined as:

$$U_i^{\varpi}(a_i, a_{-i}) = \sum_{j \in N_i^{\varpi}} \frac{(1-\rho_j(a_i, a_{-i}))^{1-\alpha}}{\alpha - 1}, \tag{3.10}$$

where $\rho_j(a_i, a_{-i})$ is the load of BS j given in (3.6), N_i^{ϖ} is the neighborhood of BS i, and ϖ is a parameter to control the nieghborhood size. The construction of neighborhood and the effect of ϖ are described in the next subsection.

In the following, we first formally define a user association game then show that it is a ξ-potential game. Note that an exact potential game is also a ξ-potential game with $\xi = 0$.

Definition 12 *[User Association Game] It is a game* $\Gamma^{\varpi} := \left\{ \mathscr{S}, \{X_i\}_{i \in \mathscr{S}}, \left\{U_i^{\varpi}\right\}_{i \in \mathscr{S}} \right\}$, *where* $\mathscr{S}$ *is a set of BSs,* X_i *is a set of strategies of BS i, and* U_i^{ϖ} *is a cost function given in (4.8). Strategy set* X_i *is a discrete set of CRE bias values for small BS* $i \in \mathscr{B}_s$ *and* X_i *is a discrete set of ABS ratios for macro BS* $i \in \mathscr{B}_e$.

Proposition 7 *The user association game* Γ^{ϖ} *with cost function (4.8) is a* ξ*-potential game with the potential function (5.3), where* ξ *is a function of* ϖ*:*

$$\xi = \max_{a_i, a_i' \in X_i, a_{-i} \in X_{-i}, i \in \mathscr{S}} \left| \sum_{j \in N_i^0 \setminus N_i^{\varpi}} \frac{\left(1-\rho_j(a_i, a_{-i})\right)^{1-\alpha}}{\alpha - 1} - \sum_{j \in N_i^0 \setminus N_i^{\varpi}} \frac{\left(1-\rho_j(a_i', a_{-i})\right)^{1-\alpha}}{\alpha - 1} \right|. \tag{3.11}$$

Proof: See Appendix 3.7.3. ■

Corollary 3 *The game* Γ^0 *is an exact potential game.*

3.3.2 Base Station Neighborhood

We start with the definition of the neighbour set N_x of small BSs at location x:

$$N_x = \left\{ j \in \mathscr{S} \,|\, \max_{c_j} P_j g_j(x) c_j \geq \max_{k \in \mathscr{S}} \min_{c_k} P_k g_k(x) c_k \right\}. \tag{3.12}$$

BS j is in N_x if it is likely to serve the user at x for some CRE vector. Take the example of Fig. 3.2, which shows the bias received power range at a given location x for all BSs. The BSs whose biased received power range intersect with the line that passes through the max-min biased received power are the neighbor BSs. In Fig. 3.2, BS 1 is a macro BS and has a single possible CRE bias. It also has the max-min bias received power. This means

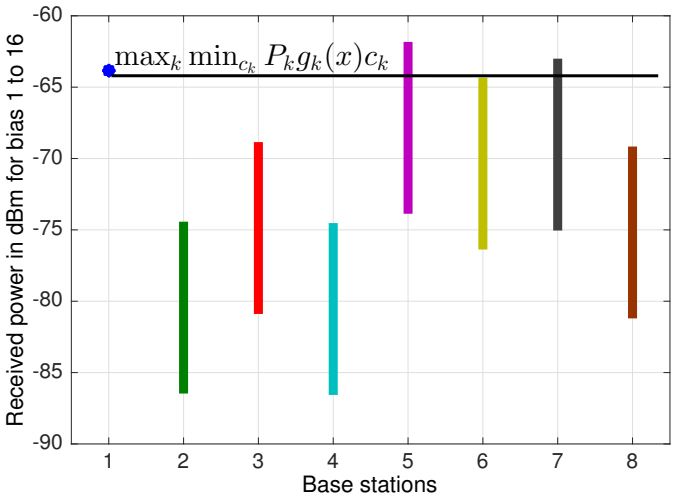


Figure 3.2: [Ilustration of the neighbour set N_x]: In this figure the biased received power range of 8 BSs at a location are shown. BS 1 is a macro BS with a single CRE bias of 1, making a single point on the figure. The BSs whose biased received power range intersect with the line that passes through the max-min biased received power are neighbours.

a user at x will receive at least this bias power. The bias received power from BS 5 can exceed this max-min, so that BS 5 is likely to serve the user for some CRE bias and is thus included in N_x. In the same way, BS 7 is also included in N_x. On the other hand, the bias received power from BS 2 will never exceed that of BS 1 and thus BS 2 will never serve the user at x.

We now construct the neighborhood of small BSs based on sets N_x. We assume that the users located at x calculate N_x and report it to their serving BS, which multicast this information to all the BSs in N_x. BS j is considered to be a neighbor of BS i if the proportion of reports where BS i and BS j are in N_x is at least a threshold ϖ, which is neighbourhood control parameter. This constraint aims at excluding from the neighborhood BSs that have insignificant influence on load. Otherwise, due to the infinite support of shadowing in our model, all BSs in the network can be potentially neighbors. In case of only pathloss model the neighborhood size is smaller because only the geographically near BSs affect each other. Formally, the neighbourhood of small BS i is defined as:

$$N_i^{\varpi} = \left\{ j \in \mathscr{S} \,|\, \frac{\int_{x \in \mathscr{L}} \lambda(x) 1_{i,j \in N_x} dx}{\int_{x \in \mathscr{L}} \lambda(x) 1_{i \in N_x} dx} \geq \varpi \right\}. \tag{3.13}$$

For the threshold $\varpi = 0$ the neighbour set N_i^0 boils down to $N_i^0 = \bigcup_{x: i \in N_x} N_x$. The neighbour set N_i^{ϖ} is empty for $\varpi > 1$. For $0 < \varpi \leq 1$, N_i^{ϖ} is a decreasing sequence of sets.

Note that a change in the ABS ratio theoretically affects the load of all BSs of the network through interference. We thus assume that a macro BS neighborhood is made of all BSs. In practice, however, the neighborhood of a macro BS is finite because interference power decreases with distance. Macro BS neighborhood can be constructed similarly to the small BS neighborhood construction as shown in Appendix 3.7.5.

Note that for the shadow fading scenario neighbourhood N_i^0 is potentially large whereas for only pathloss scenario N_i^0 is smaller. Therefore, to reduce of the size of neighborhood in the shadow fading scenario we use $N_i^{\varpi} \subset N_i^0$ that results into a near-potential game Γ^{ϖ} as proved in Proposition 7. Whereas, for pathloss only scenario we use N_i^0 that result into an exact potential game Γ^0 as given in Corollary 3.

3.4 Distributed Learning Algorithms

In this section, we adapt the developed distributed BLLA for the ξ-potential game Γ^{ϖ} to find the optimal of the potential function ϕ_{α}. First, we present the adaptation of BR algorithm and the LLA for the complete information setting. Next, the BLLA for the partial information setting is described.

3.4.1 Best Response Algorithm

Best Response (BR) algorithm is an asynchronous algorithm where at any given time only a single BS updates its strategy. Set ϖ and assume a time-varying random process with which a BS is chosen to revise its strategy[3]. The selected BS computes its cost $U_i(a_i, a_{-i}(t-1))$ for all $a_i \in X_i$ and sets $U_i(a_i, a_{-i}(t-1)) = \infty$ if $\rho_j \geq 1$ or $O_j \geq \bar{O}_j$ for $j \in N_i^{\varpi}$. Then, the BS chooses a strategy $a_i \in X_i$ that minimizes its cost, given the strategies $a_{-i} \in X_{-i}$ of other players. In other words, BS i chooses a strategy from its best response set B_i:

$$B_i(a_{-i}) = arg\,min_{a_i} U_i(a_i, a_{-i}). \tag{3.14}$$

We note that the BR algorithm requires complete information, i.e., the effects of choosing all the other strategies are supposed to be known. Moreover, it is known that BR algorithm is not guaranteed to converge to the optimal PNE even in exact potential game Γ^0 because the potential function may have multiple sub-optima [3]. For the ξ-potential

[3]Uniform probability or stationarity of the process is not required, it is only required that the probability of selecting any player is positive.

game Γ^{ϖ} ($\varpi \neq 0$) and $\xi \neq 0$, a PNE may not even exist. Therefore, we adapt LLA for the ξ-potential game Γ^{ϖ}.

3.4.2 Log-linear Learning Algorithm

The adaptation of LLA is summarized in Algorithm 4. However, for this algorithm also, the BSs require complete information. For example, given the strategies of others, the BS has to know the cost function value for all its strategies to compute (3.15). With this information, it selects a strategy to play according to a probability distribution. In general, acquiring this amount of information is not feasible. To overcome this difficulty in the next subsection we propose to use BLLA.

Algorithm 4 Log-linear Learning Algorithm

1: **Initialisation:** Start with arbitrary action profile $a(0)$.
2: Set parameter τ and ϖ.
3: **While** $t \geq 1$ **do**
4: Randomly a BS i is selected to revise its action.
5: Compute cost $U_i(a_i, a_{-i}(t-1))$ for all $a_i \in X_i$.
6: For any $a_i \in X_i$, set $U_i(a_i, a_{-i}(t-1)) = \infty$ if $\rho_j \geq 1$ or $O_j \geq \bar{O}_j$ for $j \in N_i^{\varpi}$.
7: Take action $a_i(t)$ from X_i with probability $p_i^{a_i}(t)$,

$$p_i^{a_i}(t) = \frac{\exp\left(-\frac{1}{\tau}U_i(a_i, a_{-i}(t-1))\right)}{\sum_{a_i' \in X_i} \exp\left(-\frac{1}{\tau}U_i\left(a_i', a_{-i}(t-1)\right)\right)}. \tag{3.15}$$

8: All the other players must repeat their previous actions, i.e., $a_{-i}(t) = a_{-i}(t-1)$.

3.4.3 Binary Log-linear Learning Algorithm

BLLA just needs the partial information, which is the information about the players current and trial actions. Unlike complete information in LLA, the effect of choosing any other strategy is not required at the BS in BLLA. BLLA is also an asynchronous algorithm[4]. In this algorithm, whenever the BS updates its action it does it in two steps. In the first step, the BS tries an action from its strategy set to obtain its cost. In the second step, the BS randomly chooses among the two actions (present and trial) as summarized in Algorithm 5.

[4] Asynchronicity is in fact only required within each neighborhood.

Algorithm 5 Binary Log-linear Learning Algorithm

1: **Initialisation:** Start with arbitrary action profile $a(0)$.
2: Set parameter τ and ϖ.
3: **While** $t \geq 1$ **do**
4: Randomly a BS i is selected to revise its action.
5: Select a trial action $\hat{a}_i \in X_i$ with uniform probability.
6: Compute cost $U_i(\hat{a}_i, a_{-i}(t-1))$.
7: Set $U_i(\hat{a}_i, a_{-i}(t-1)) = \infty$ if $\rho_j \geq 1$ or $O_j \geq \bar{O}_j$ for $j \in N_i^{\varpi}$.
8: Play action $a_i(t) \in \{a_i(t-1), \hat{a}_i\}$ as given below.

$$a_i(t) = \begin{cases} a_i(t-1), & \text{w.p. } \frac{e^{-\frac{1}{\tau}U_i(a(t-1))}}{e^{-\frac{1}{\tau}U_i(a(t-1))}+e^{-\frac{1}{\tau}U_i(\hat{a}_i,a_{-i}(t-1)))}}, \\ \hat{a}_i, & \text{w.p. } \frac{e^{-\frac{1}{\tau}U_i(\hat{a}_i,a_{-i}(t-1)))}}{e^{-\frac{1}{\tau}U_i(a(t-1))}+e^{-\frac{1}{\tau}U_i(\hat{a}_i,a_{-i}(t-1)))}}. \end{cases} \tag{3.16}$$

9: All the other BSs must repeat their previous actions, i.e., $a_{-i}(t) = a_{-i}(t-1)$.

Note that in all the above algorithms the actions that are not in the feasible set $\mathscr{F}$ have infinite cost. If there is no action in the feasible set $\mathscr{F}$ at time t then by convention we set CRE bias equal to unity for small BSs and ABS ratio equal to zero for macro BSs.

The proof of convergence of LLA and BLLA to optimal PNE for an exact potential game Γ^0 is given in [20]. We have shown in Theorem 1 of Chapter 2 that under certain conditions both of these algorithms converges to optimal PNE for the ξ-potential game. We derive sufficient conditions in terms of the system parameter so as to meet the condition of Theorem 1.

Let ϕ_α^* and $\phi_\alpha^\dagger$ be the first minimum and second minimum values of the potential. The following theorem gives an upper bound on ϖ that guarantees LLA and BLLA to converge to an optimal PNE.

Theorem 8 *The constraint in Theorem 1 is satisfied if*

$$\varpi \leq \varepsilon Q (1-\rho_{\max})^{\alpha}, \tag{3.17}$$

where $\rho_{\max}$ *is the maximum possible load of a BS,* $Q = \frac{\max_{x,\bar{\theta},j\in\mathscr{S}} \frac{1}{\nu_j(x,\bar{\theta})}}{4|\mathscr{S}||X|\lambda_m \max_x\left\{\frac{1}{\mu(x)}\right\}}$, *and* λ_m *is an upper bound for the sum arrival rate in a cell.*

Proof: See Appendix 3.7.4 ■

Corollary 4 *The constraint in corollary 1 is satisfied if*

$$\varpi \leq Q\left(\phi_{\alpha}^{\dagger} - \phi_{\alpha}^{*}\right)(1-\rho_{\max})^{\alpha}. \tag{3.18}$$

Remark Recall that we have assumed that the neighborhood of every macro BS is made of all the BSs in the network. It is however also possible to restrict it by following a similar technique as for small BSs. The definition (3.13) should be extended by considering the set of macro BSs that significantly affect the load of BS i. Then, similar conditions for convergence of LLA and BLLA can be obtained. See Appendix 3.7.5 for more details.

3.4.4 Effect of time-varying neighbours

For all the above algorithms, the BS i needs to know its neighbours N_i^{ϖ} to calculate its cost. Providing to every BS the neighbors set is a standard task for network operators for a given CRE bias vector. This task can be performed automatically e.g., using automated neighbor relation (ANR) standardized by 3GPP [70]. The difficulty here is to determine N_i^{ϖ} that depends on all the possible values of the CRE bias vector (see (3.13)). To address this problem, we propose the following technique.

At every packet call the user at location x calculates N_x and reports it to its serving BS, which multicast this information to all the BSs in N_x. In the process of learning, BS i updates N_i^0 whenever it receives reports from other BSs. From this, the BS calculates the proportion of reports of neighbor BSs. If the proportion of reports of BS j exceeds the threshold ϖ then it is included in N_i^{ϖ}. Since, the shadowing and the traffic is stationary the estimate of proportion of reports converges over time. Hence, N_i^{ϖ} $\forall i$, also converges. Thus, the algorithms LLA and BLLA converge to the global minimum of the objective function as proved in Corollary 1.

3.5 Simulation Results

In this section, we show simulation results of load balancing in heterogeneous networks using the proposed algorithms. We first consider only pathloss model and use CRE bias for load balancing. Next, we consider shadow fading and use CRE bias for load balancing. Next, in the same scenario, we use both CRE bias and ABS to clearly show the effect of ABS on load balancing.

We consider standard parameters as adopted in 3GPP [71]. These parameters are listed

Table 3.1: Simulation parameters.

Parameter	Variable	Value
Number of BSs	N_s	8
Macro BS during NS	P_{macro}	46 dBm
Macro BS during ABS	P_{ABS}	0 dBm
Small BS	P_{small}	24 dBm
Average file size	$\frac{1}{\mu}$	0.5 Mbytes
Average traffic load density	$\frac{\lambda}{\mu}$	64 bits$/s/m^2$
System bandwidth	W	20 MHz
Carrier frequency	f_c	2.6 GHz
Noise power	N	-174+10log(W) dBm
Minimum SINR	γ_{min}	-10 dB
pathloss exponent	η	3.5
Reference distance	d_0	10 m
CRE bias set	c_i	$\{1, 1.1, 1.2, \ldots, 16\}$
ABS ratio	θ_i	$\{0, 0.01, 0.02, \ldots, 1\}$

in the Table 3.1. We consider 8 BSs located in a two dimensional region $\mathscr{L}$. BS 1 is a macro BS that transmits with P_{macro} and the rest are small BSs that transmit with P_{small}. The user traffic varies with location across an average traffic density of 64 bits$/s/m^2$. There are two hotspots where the traffic is 5 times the average traffic, which can be seen in Fig. 3.3. We use the classical Shannon formula for calculating channel capacity $\tilde{v}_i^f(x, \bar{\boldsymbol{\theta}}) = W \log_2\left(1 + \gamma_i^f(x, \bar{\boldsymbol{\theta}})\right)$.

3.5.1 Load Balancing with CRE Bias in Pathloss Scenario

In this section, we consider pathloss scenario to show that in this scenario exact potential game Γ^0 can be used for load balancing. The exact potential game Γ^0 is suitable in this scenario because it is obtained by using the neighborhood N_i^0, which in this scenario is smaller due to the absence of shadow fading. We use only CRE bias for load balancing in this scenario.

We compare the convergence performance of BR, LLA and BLLA in Fig. 3.4. We observe that BR converges much faster than the LLA and the BLLA. However, BR get stuck into a local minimum. Similarly, with a small $\tau = 0.001$ LLA get stuck into the same local minimum where BLLA escapes it. For a higher $\tau = 0.01$ LLA and BLLA achieves better minimums. BLLA requires only partial information, which is an advantage for practical implementations but does not loose much in terms of convergence speed.

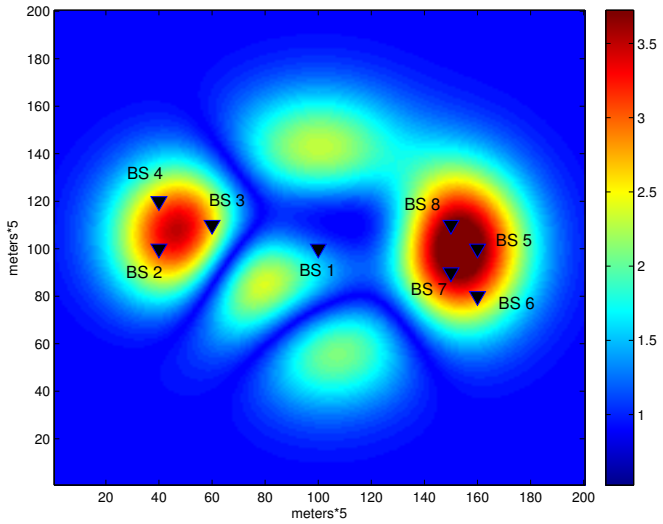


Figure 3.3: This figure shows a normalised traffic variations in a square kilometer region. There are 8 BSs, where BS 1 is macro and the rest are small BSs that are located around two hot spots of traffic.

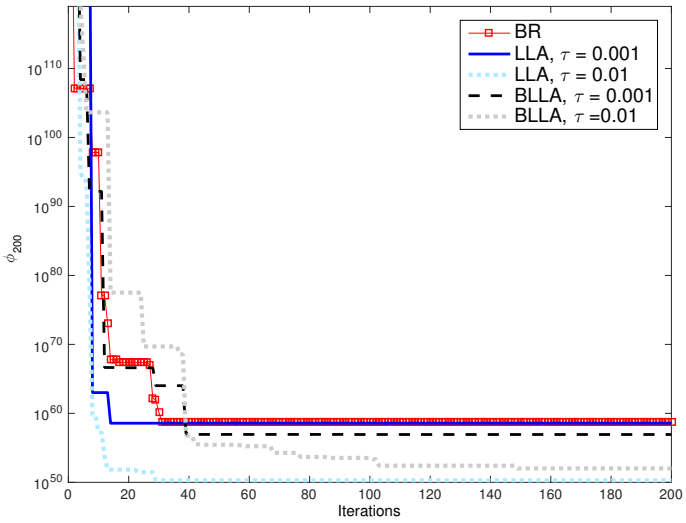


Figure 3.4: [($\alpha = 200$) Min-max policy]: This figure shows the convergence of BR, LLA, and BLLA. We see that the BR converges fast to a local minimum. Also, with small $\tau = 0.001$, LLA get stuck into the same minimum. Both LLA and BLLA attain a better minimum with $\tau = 0.01$. It can be seen that due to partial information BLLA takes a longer time than LLA.

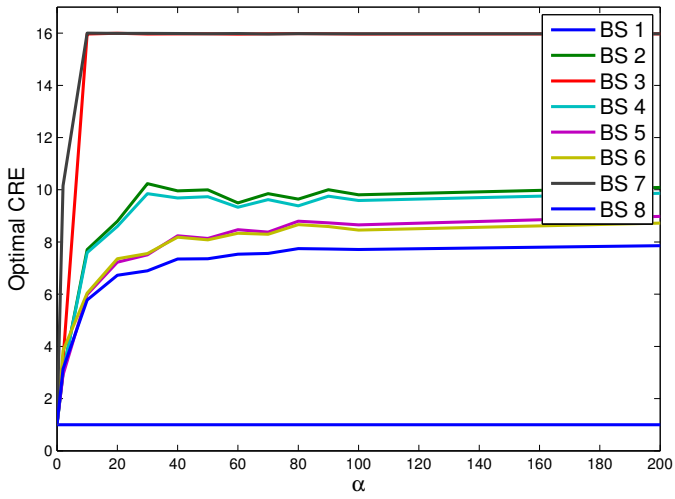


Figure 3.5: [Evolution of optimal CRE bias vector with α under pathloss scenario]: For $\alpha = 0$, the CRE bias of all the BSs is one which represent the conventional user association without bias. As α increases the CRE biases of BSs also increases to balance the loads of BSs. Macro BS 1 does not have CRE bias, hence its bias is always unity.

Now, using LLA, we study the evolution of CRE bias vector and optimal loads by varying α. The evolution of the optimal CRE biases and optimal loads for α values ranging from 0 to 200 is shown in Fig. 3.5 and Fig. 3.6. These figures shows how the optimality changes from rate-optimal to min-max optimal as α increases from 0 to 200. Therefore, initially at $\alpha = 0$, the optimal bias of all BSs is close to one and correspondingly there is a heavy load imbalance. As α increases, the load of the macro BS decreases and that of the small BSs increases. At around $\alpha = 50$ the min-max load policy is reached. It is observed that different BSs have different optimal bias values for different α.

We study the optimal CRE bias vector and optimal coverage regions of BS in the following. In Table 3.2, we show the obtained optimal CRE bias vector and optimal loads for different values of α. For $\alpha = 0$, a rate-optimal policy is obtained. The optimal CRE bias values of all the small BSs is close to one. This is intuitive because for rate-optimal policy the bias values should be equal to one. This case corresponds to the classical user association without the use of CRE bias leading to a heavy load imbalance. We can observe that the load of the macro BS 1 is 92%, which is near an overload. On the other hand, the loads of all the small BSs are less than 12%, which is a heavy under-utilisation. This case serves as a benchmark for other cases of α to compare for load balancing.

As α increases to 2 the optimal CRE biases increase. The load of the macro BS is decreased to 62% and the utilisation of small BSs is increased. In Fig. 3.7, the coverage

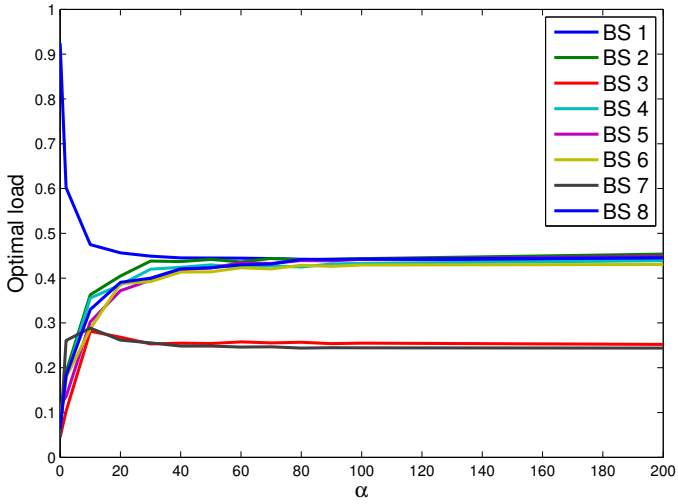


Figure 3.6: [Evolution of optimal loads of BSs with α under pathloss scenario]: For $\alpha = 0$, macro BS 1 is heavily loaded 92% while the loads of small BSs is less than 12%. As α increases the load of macro BS decreases and that of small BSs increases, finally attaining a min-max loads vector. Loads of BSs 3 and 7 does not increase more because they are located near to macro BS whose interference effect the load of these BSs.

regions for the min-max policy, which is obtained for $\alpha = 200$, is shown. When α increases to higher value, here $\alpha = 200$, the load of the macro BS 1 is further reduced to 42%. It can be observed that the loads of all the BSs are equalised except for BS 3 and 7. The utilisation of these BSs cannot be increased further because these BSs are near to the macro BS 1 that causes heavy downlink interference to their users.

Table 3.2: Comparison of optimal CRE, optimal loads of BSs for different α.

	$\alpha = 0$		$\alpha = 2$		$\alpha \to \infty$	
BS i	c_i^*	$\rho_i^*\%$	c_i^*	$\rho_i^*\%$	c_i^*	$\rho_i^*\%$
1	1	92	1	62	1	42
2	1.1	9	3.1	20	16	51
3	1	4	3.6	11	16	21
4	1	7	2.8	17	14.8	49
5	1.1	12	3.4	23	7.7	42
6	1.1	8	3.4	20	7.7	41
7	1.1	5	3.5	12	16	25
8	1	6	3.2	18	7.2	42

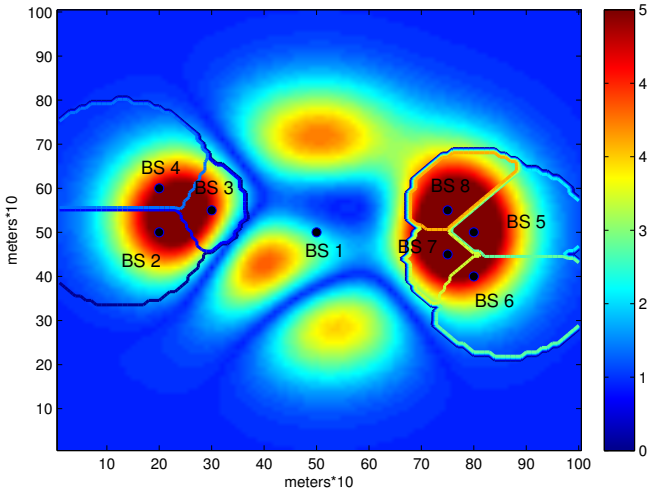


Figure 3.7: [($\alpha = 200$) Min-max policy:] Coverage region of BSs are shown with boundaries under pathloss. These regions are obtained with the optimal CRE biases. It can be seen that the coverage regions of BSs 3 and 7 are smaller that result in smaller loads of these BSs.

3.5.2 Load Balancing with CRE Bias in Shadow Fading

In this section, shadow fading scenario is considered. The neighbhourhood of BSs becomes larger due shadow fading as discussed in Section 3.3.2. We use the near-potential game Γ^{ϖ} for this scenario as it allows to use the reduced neighbourhood N_i^{ϖ}. We consider a shadow fading of standard deviation of $\sigma_{sh} = 8$ dB and a decorrelation distance of $D_c = 20$ m. Cross-correlation between the shadowing components at a location is considered to be 0.5. We first focus on the CRE optimization and assume in this section that the macro BS does not implement ABS ($\theta_1 = 0$). We consider a square region $\mathscr{L}$ of side 1000 m.

The convergence of LLA and BLLA using the parameter values $\tau = 10^{-3}$, $\varpi = 10^{-22}$ is shown in Figures 3.8, 3.9, 3.10 for $\alpha = 0,\ 2,\ 50$, respectively. We observe that in all these figures proposed algorithms converge within few tens and sometimes few hundreds of iterations. We also see that LLA converges faster than BLLA. This is of course due to the complete information setting assumed for LLA. Note however that BLLA does not lose so much in terms of convergence speed; this is an interesting conclusion for a practical implementation.

We also compare LLA and BLLA with Log-linear Trial Error (LLTE) learning algorithm, which is completely uncoupled and a variant of trial and error algorithm [72]. It guarantees asymptotic convergence to the optimal NE for any finite game that possesses at least one

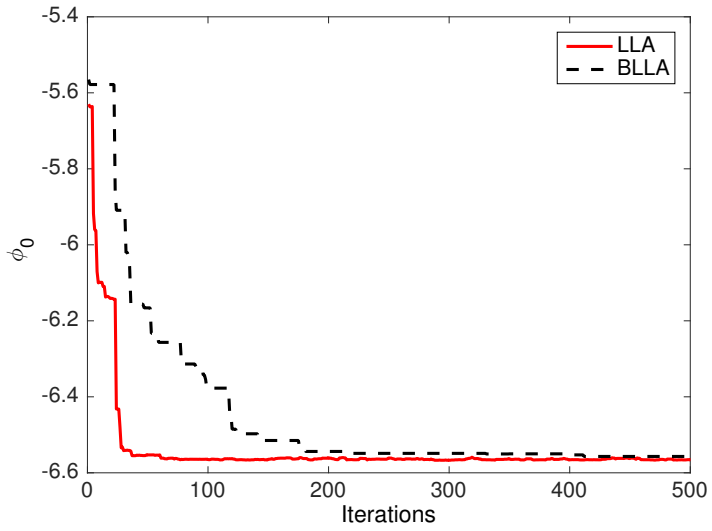


Figure 3.8: [$(\alpha = 0)$ Rate-optimal policy]: This figure compares the convergence of LLA and BLLA using $\tau = 0.001$ and $\varpi = 10^{-22}$. In this case, the objective function is negative. Both LLA and BLLA converges to the same minimum. LLA converges within a few tens of iterations while BLLA takes more iterations because it uses partial information.

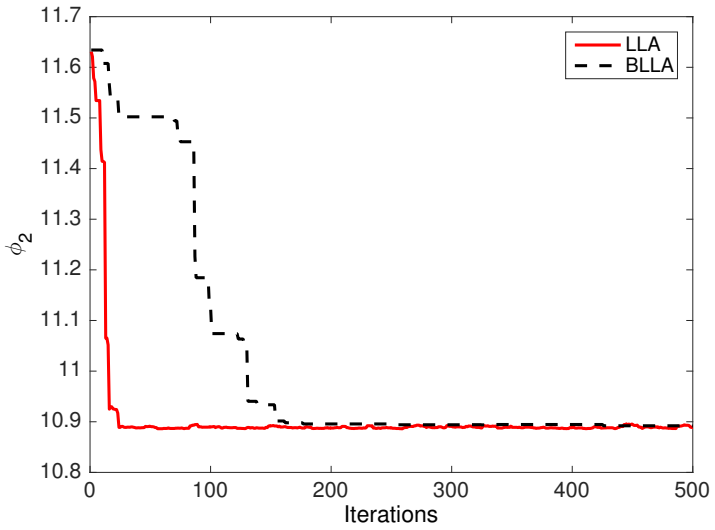


Figure 3.9: [$(\alpha = 2)$ Delay-optimal policy]: This figure compares the convergence of LLA and BLLA using $\tau = 0.001$ and $\varpi = 10^{-22}$. In this case, the objective function is positive and of the order of 10. Both LLA and BLLA converges to the same minimum. Here also, LLA converges faster than BLLA.

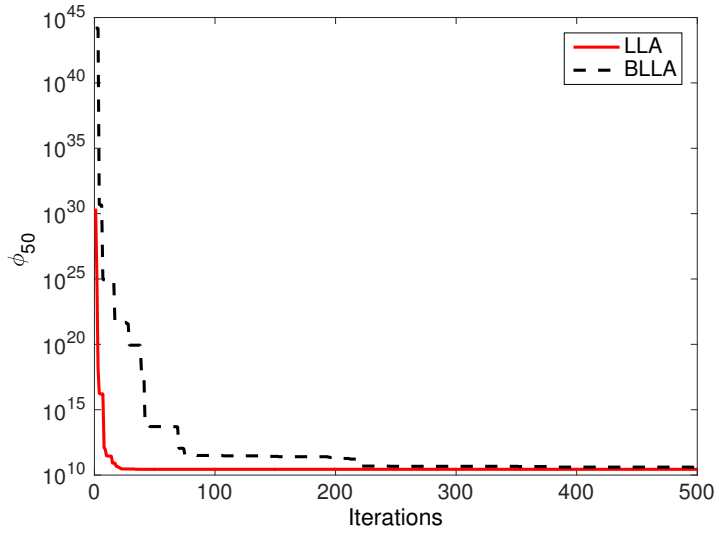


Figure 3.10: [($\alpha = 50$) Min-max policy]: This figure compares the convergence of LLA and BLLA using $\tau = 0.001$ and $\varpi = 10^{-22}$. In this case, the objective function values are exponentially large. Both LLA and BLLA converges to the same minimum. BLLA takes more iterations than LLA.

NE. Hence, it is a suitable algorithm to compare with LLA and BLLA for near-potential game Γ^{ϖ}. The comparison of evolution of objective function ϕ_{50} is shown in Fig. 3.11 [5]. LLA and BLLA converge quickly whereas LLTE oscillates between fast search and slow search phases. Therefore, the performance of LLA and BLLA for load balancing is much superior to that of LLTE.

Effect of ϖ

The effect of threshold ϖ on the convergence of LLA is shown in Fig. 3.12 for $\tau = 10^{-3}$ and for the particular case $\alpha = 50$ (results for BLLA and other values of α provide similar conclusions). For $\varpi = 0$ all the BSs are neighbors so that our framework is an exact potential game and LLA converges to an optimal PNE. Threshold parameter $\varpi = 10^{-22}$ results in an ξ-potential game and satisfies the sufficient condition of Corollary 4. Therefore, LLA converges also to the global minimizer of the objective function. Although small, this value of ϖ significantly shrinks the neighborhood set of the BSs. In the scenario of Fig. 3.3, BSs 2 and 4 are not anymore neighbours of BSs 5, 6, and 7. If we now further increase ϖ to a value that violates the condition of Corollary 4 ($\varpi = 0.9$), neighborhoods are further

[5]The exploration parameter ε of LLTE is set to $\varepsilon = 0.1$. The functions that govern slow and fast search phases LLTE are chosen to satisfy the conditions [72, (1), (2), and (3)].

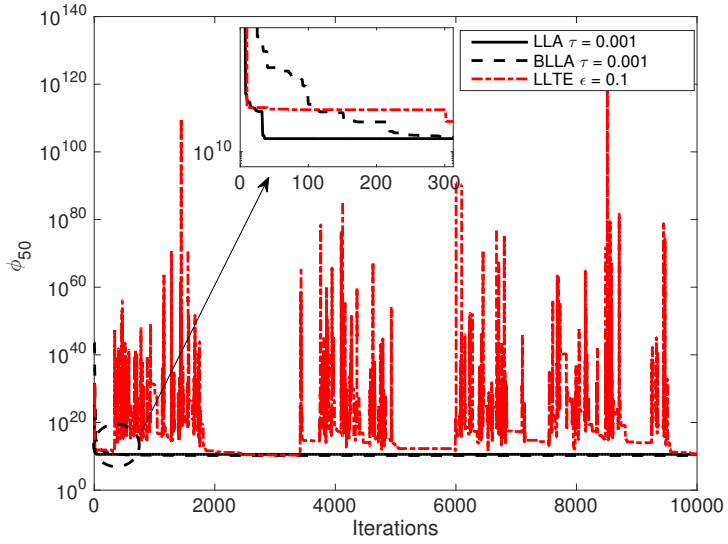


Figure 3.11: In this figure, we compare LLA, BLLA using $\alpha = 50$, $\varpi = 10^{-22}$ with LLTE using the exploration parameter $\varepsilon = 0.1$. We see that the LLA and BLLA converges within few iterations while LLTE algorithm oscillates.

reduced. In our scenario, BS 1 is, for example, the only neighbor of BSs 3, 6, and 8. LLA is however not anymore guaranteed to converge to an global minimum as seen from the figure. The threshold ϖ therefore strikes a balance between the size of the neighborhood and the optimality of the solution.

Effect of τ

There is a trade-off in the choice of τ. LLA and BLLA converge with high probability to the global minimum of the objective function for $\tau \in (0, \infty)$ under the conditions of Corollary 1. This means that, asymptotically, the probability that the algorithm is at the global minimum approaches to one as τ goes to zero.

For high values of τ, LLA and BLLA results into oscillations. This is due to the fact that the algorithms converge fastly in probability to the uniform distribution. As a mater of fact, it doesn't spend much time in optimal states, which is not practically desirable. For small values of τ, asymptotically, the algorithms will spend most of the time in the global optimal. However, convergence is slow in probability. This explains that the system can take a long time to escape from sub-optimal states. Contrary to best response, however, the proposed algorithms will not get stuck into these sub-optimal states.

The effect of τ on the convergence of LLA and BLLA for $\alpha = 2$ and $\varpi = 10^{-22}$ is

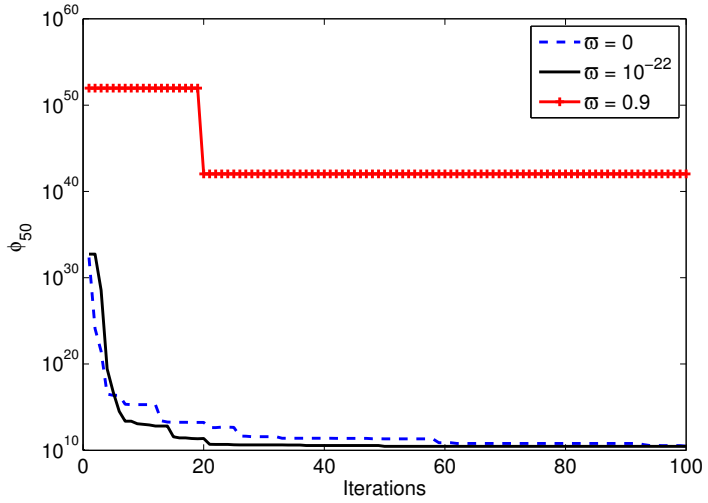


Figure 3.12: [Effect of ϖ on LLA with fixed $\tau = 10^{-3}$ and $\alpha = 50$]: For $\varpi = 0$, all the BSs are neighbours that makes the user association game an exact potential game. Hence, LLA converges to the global minimum. However, for $\varpi = 10^{-22}$ that satisfy Corollary 4, the neighbourhood shrinks. In Fig. 3.3, BSs 2 and 4 are not anymore neighbours of BSs 5, 6, and 7. This makes the user association game a ξ-potential game. Hence, LLA also converges to the same global minimum. Whereas, for $\varpi = 0.9$ that does not satisfy Corollary 4, LLA does not converge to global minimum.

shown in Fig. 3.13 and Fig. 3.14, respectively. For high value of $\tau = 0.05$, LLA results into oscillations. It reaches the global minimum (around iteration 300) but does not spend much time in the optimum states. For $\tau = 0.01$, the time spent in the optimum states is increased. For a carefully chosen $\tau = 0.001$, the system spends most of the time in the optimal states. We can see in Fig. 3.14 that the behavior of BLLA is similar to that of LLA, except that it takes more iterations to hit the global minimum of the potential function because of the partial information used.

Effect of α

We now compare in Table 4.2 the optimal bias values and BS loads. For $\alpha = 0$ the coverage region is shown in Fig. 3.17. Every user is served by the BS that provides maximum data rate, which is obtained for bias values equal to one. This corresponds to the classical best signal association rule that results in heavy load imbalance between stations: the load of the macro BS reaches 92%, while small BSs have loads less than 11%.

As α increases to 2 the coverage regions of all small BSs shown in Fig. 3.18 expands and that of the macro BS shrinks. The load of the macro BS is decreased to 61% and

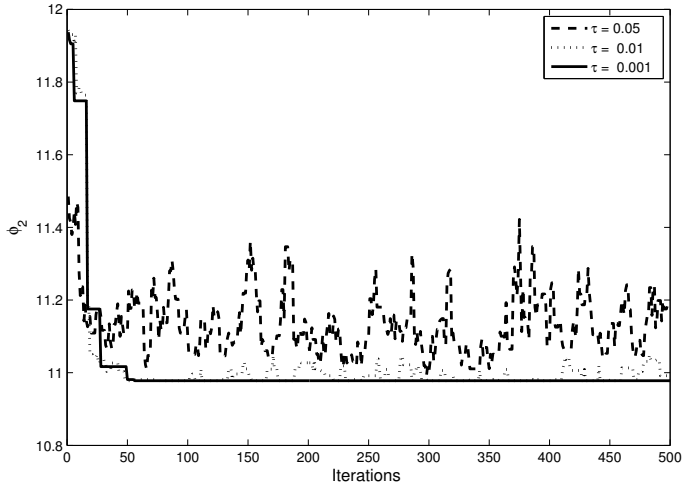


Figure 3.13: [Effect of τ on LLA]: Large $\tau = 0.05$, results into higher oscillations of LLA compared to that of $\tau = 0.01$. A carefully chosen $\tau = 0.001$ gives the desired convergence to global minimum.

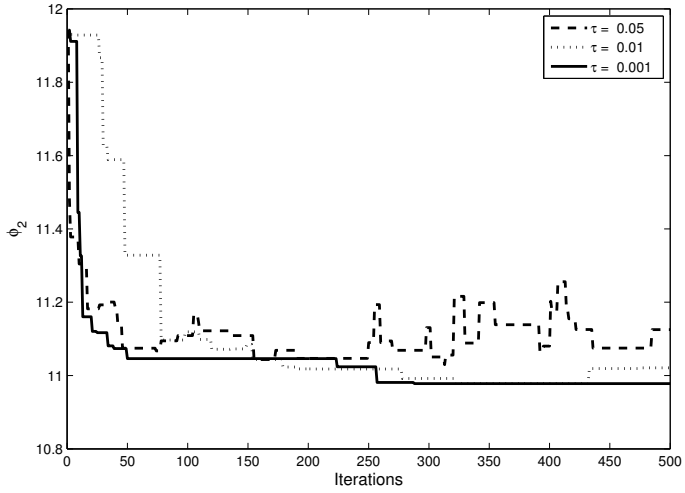


Figure 3.14: [Effect of τ on BLLA]: The $\tau = 0.001$ converges to the global minimum compared to that of other values of τ similar to as for LLA. However, it takes more iterations compared to that of LLA.

Table 3.3: Comparison of optimal CRE, optimal loads of BSs for different α ($\tau = 10^{-3}$, $\varpi = 10^{-22}$).

	$\alpha = 0$		$\alpha = 2$		$\alpha \to \infty$	
BS i	c_i^*	$\rho_i^*\%$	c_i^*	$\rho_i^*\%$	c_i^*	$\rho_i^*\%$
1	1	92	1	61	1	45
2	1	7	3	20	8	42
3	1	4	3	9	9	23
4	1	9	3	18	8	37
5	1	11	3	21	7	37
6	1	8	3	20	7	43
7	1	5	3	11	8	30
8	1	7	3	19	6	37

concurrently the utilization of small BSs is increased (up to 21%). Min-max policy is approximated with a value of $\alpha = 50$, coverage region is shown in Fig. 3.19. The load of the macro BS is further reduced to 45% and load dispersion is decreased.

This phenomenon can also be observed in Figures 3.15 and 3.16, where optimal CRE and loads are shown as functions of α. Every point on this figure is obtained by averaging over 50 realizations of LLA. By definition, the CRE of the macro BS is constant and equal to 1. We see on this figure how the CREs of small BSs are gradually increased and how the load dispersion is reduced. We also conclude that choosing $\alpha = 50$ provides a good approximation for the min-max policy in this scenario.

3.5.3 Load Balancing with CRE and ABS in Shadow Fading

In this section, we allow the macro BS to implement ABS and study the effect of this technique on load balancing. For the sake of simplicity, we set $\tau = 10^{-3}$, $\varpi = 10^{-22}$ and use LLA. We also consider a square region $\mathscr{L}$ of side 2000 m in the simulations. Three cases may be compared to evaluate the interest of using ABS:

1. *No outage constraint no ABS:* In this case, the macro BS does not implement ABS ($\theta_1 = 0$) and we do not impose outage constraint. This serves as a benchmark to compare with other two cases.

2. *Outage constraint without ABS:* We introduce here an outage constraint ($\bar{O}_i = 2\%$ for all i) but still don't allow ABS ($\theta_1 = 0$). The outage is taken into account by setting an infinite cost to actions violating the constraint $\bar{O}_i$ as shown in Algorithm 4.

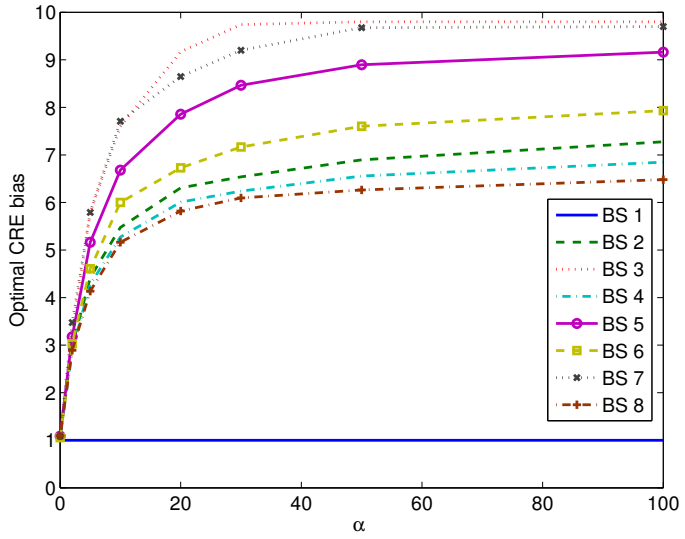


Figure 3.15: [Evolution of optimal CRE bias vector with α under shadow fading scenario]: As we have seen before, for $\alpha = 0$, the CRE bias of all the BSs is one. Compared to Fig. 3.5, the evolution of biases is smoother because of shadow fading.

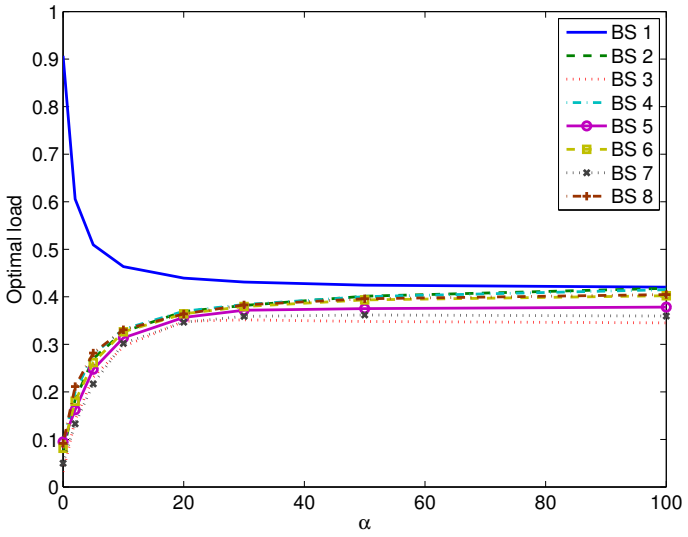


Figure 3.16: [Evolution of optimal loads of BSs with α under shadow fading scenario]: As before, for $\alpha = 0$, macro BS 1 is heavily loaded while the loads of small BSs smaller. As α increases the load of macro BS decreases and that of small BSs increases, finally attaining a min-max loads vector. Loads of BSs 3 and 7 are closer to other BSs loads because of shadow fading compared to that of in Fig. 3.6.

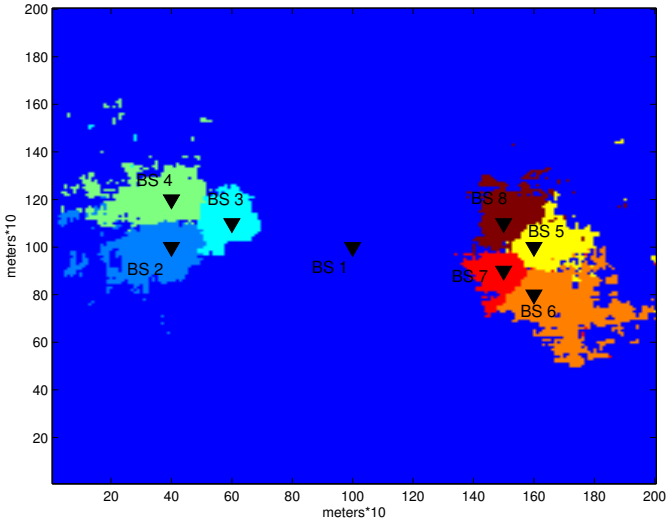


Figure 3.17: [$(\alpha = 0)$ Rate-optimal policy]: This figure shows the coverage regions of BSs obtained with the optimal CRE bias vector under shadow fading. The coverage regions are small but irregular because of shadow fading compared the fixed bounded regions in pathloss scenario in Fig. 3.7.

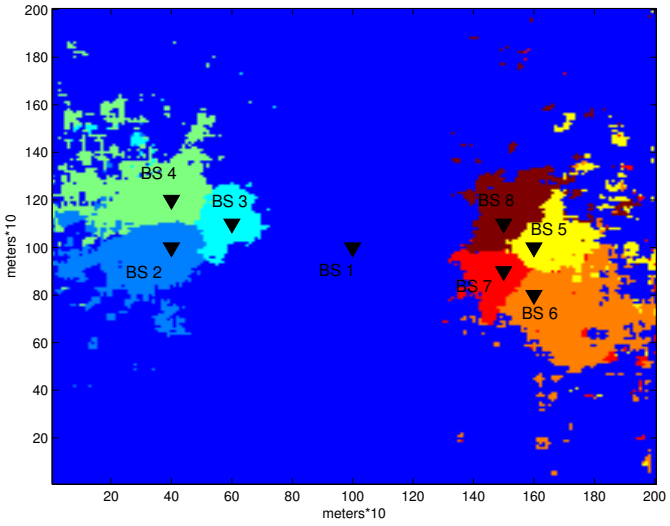


Figure 3.18: [$(\alpha = 2)$ Delay-optimal policy]: In this figure the coverage regions of BSs obtained with the optimal CRE bias vector under shadow fading are bigger compared to that of Fig. 3.17. This is because the optimal CRE biases are larger in this case.

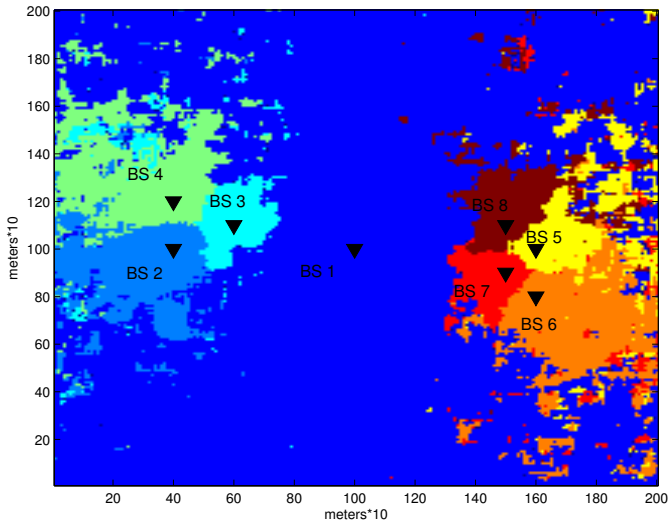


Figure 3.19: [($\alpha = 50$) Min-max policy]: This figures shows the coverage regions of BSs obtained with the optimal CRE bias vector under shadow fading. The coverage region of macro BS decreases while the coverages of small BS increases, which decreases the load of macro and increases the loads of small BSs attaining a min-max load policy.

3. *Outage constraint with ABS:* We impose an outage constraint and allow ABS at the macro BS.

We plot the cost function ϕ_α for the three considered cases in Figures 3.20, 3.21, 3.22. Each curve of the figures is obtained by averaging over 50 realisations. In our scenario, choosing $\alpha = 0$ (Fig. 3.20) results in a low outage probability. Therefore, ϕ_α curves for the first two cases are very close to each other. Since outage constraint is already met without ABS, introducing ABS does not bring any additional interest. With $\alpha = 2$ (Fig. 3.21), outage probability increases but remains below the acceptable threshold. As a consequence, the three curves also converge to the same value. With $\alpha = 50$ (Fig. 3.22), outage probabilities exceed the threshold in the first case. When outage constraint is introduced in the second case, the cost increases because the feasible set shrinks (some actions are not anymore available). When ABS is introduced in the third case, the feasible set expands again and so the optimal cost of the system decreases, fairness is improved.

Fig. 3.23 shows the evolution of the outage probabilities as the algorithm iterates (we assume $\alpha = 50$ and 50 realisations are averaged). With no ABS and no constraint (Fig. 3.23 (i)), outage probabilities of BS 3 and 7 considerably exceed the threshold of 2%. The reason is that small BSs increase their CRE to achieve optimality without taking care of users

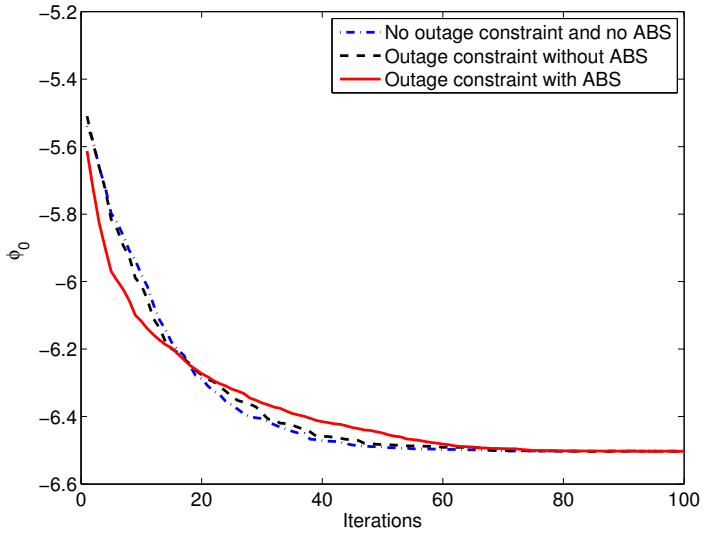


Figure 3.20: [Effect of outage constraint and ABS on load balancing for $\alpha = 0$]: Since in this case the outage constraint is met without ABS there is no difference in optimal global cost. This is because the optimal bias of all BSs are one.

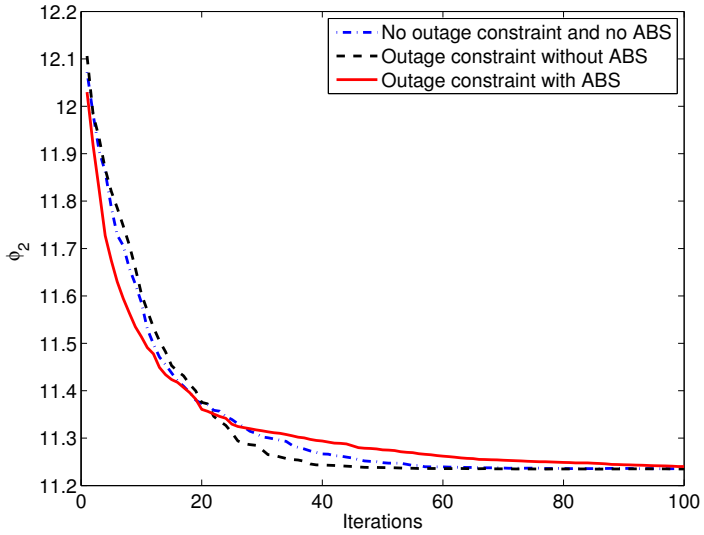


Figure 3.21: [Effect of outage constraint and ABS on load balancing for $\alpha = 2$]: As before, in this case also the outage constraint is met without ABS there is no difference in optimal global cost. This is because the optimal bias of all BSs are smaller.

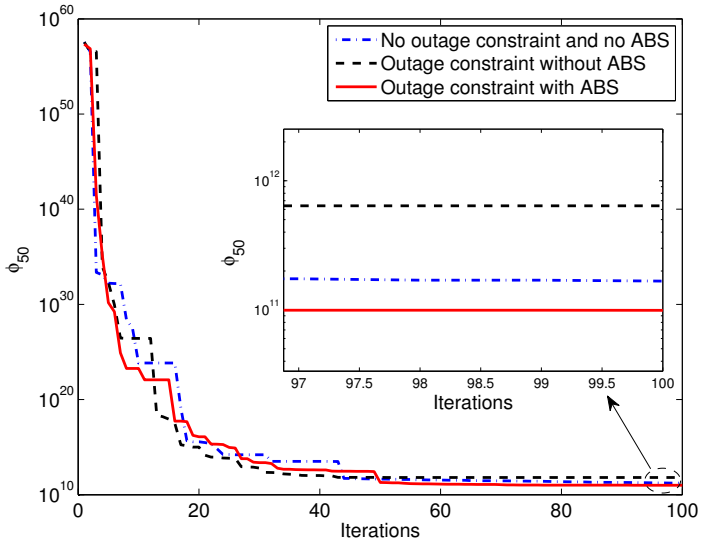


Figure 3.22: [Effect of outage constraint and ABS on load balancing for $\alpha = 50$]: There is a huge difference in optimal global cost in this case because the optimal bias of all BSs are higher that results into higher outages. We can see that the global cost is much smaller with ABS in outage constraint compared to that of without ABS.

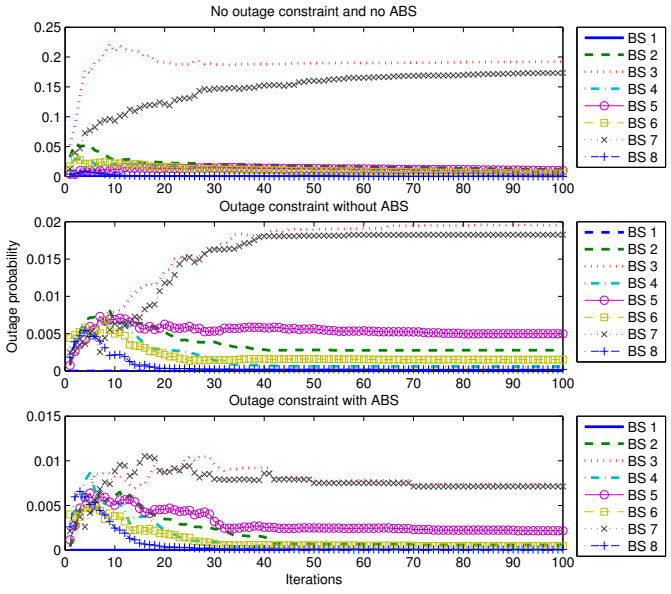


Figure 3.23: [Outage probability comparison]: We can see that without outage constraint and ABS, the measured maximum outage probability is more than 20%. When outage constraint of $\bar{O}_i = 2\%$ is introduced then the measured outage probability is under 2%. With ABS the outage probability further reduces to 1%.

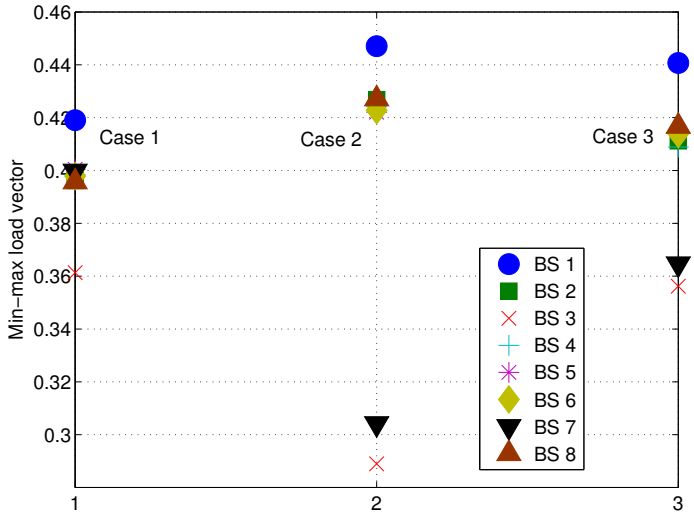


Figure 3.24: [Comparison of min-max load vector]: We observed that without outage constraint and ABS in case 1 the loads of BSs are closer. However, this happens at the cost of high outages as we have seen in Fig. 3.23. When outage constraint of $\bar{O}_i = 2\%$ is introduced in case 2 the loads have large deviations. This is because the outage constraint is met at the cost of load balancing. When ABS is introduced along with outage constraint in case 3 a better load balancing is obtained meeting the outage constraint seen in Fig. 3.23.

in outage, so that users at cell edge may experience a very bad signal quality. Therefore, fairness is achieved at the cost of an unacceptable outage. Imposing an outage constraint without using ABS is sufficient to achieve a good quality of service (Fig. 3.23 (ii)). The function ϕ_α however converges to a higher value (see Fig. 3.22). ABS is a good means to both meet the outage constraint and achieve fairness (Fig. 3.23 (iii)). The reason is that small BSs cell edge users experience a better signal quality during ABS subframes and the ABS ratio offers also the macro BS an additional degree of freedom for adapting its load and achieving fairness. This can also be seen in Fig. 3.24, where we have plotted average loads over 50 realisations after LLA has converged. From the first to the second case, the load vector expands because of the smaller feasible set and then shrinks in the third case thanks to ABS.

3.6 Conclusions

In this chapter, a novel approach for load balancing using CRE bias association technique and ABS interference management technique is presented. Our approach exploits the near-potential game structure and distributed learning algorithms. We showed that unlike in the literature the load balancing problem can be solved distributively by restricting the number of neighbours. We provide sufficient conditions under which learning algorithms converges to the optimal PNE. By running extensive simulations in two settings, which are complete and partial information settings, we show that the proposed algorithms converge within a few tens of iterations to the optimal PNE, which is also a minimiser of a $\alpha-$fairness function of the network. The convergence speed of the BLLA that uses partial information is comparable to the LLA that uses complete information, meaning that partial information is sufficient in practical implementations. Simulations showed that for load balancing LLA and BLLA perform better than a variant of trial and error algorithm. We showed that by introducing ABS the outages can be reduced and a better load balancing can be achieved. In the scenario of pathloss an exact potential game model can be used as the neighborhood is small.

3.7 Appendix

3.7.1 Proof of Rate-optimal Policy

Proof: Minimising $\phi_0(\bar{\boldsymbol{c}}, \bar{\boldsymbol{\theta}})$ is equivalent to minimise the arithmetic mean of the BSs loads, i.e., $\sum_{i\in|\mathscr{S}|}\rho_i(\bar{\boldsymbol{c}}, \bar{\boldsymbol{\theta}})$. We now prove that minimising $\phi_0(\bar{\boldsymbol{c}}, 0)$ is indeed rate-optimal policy by contradiction. Consider that at the minimum ϕ_0^* a location x_0 is associated with a BS j with rate $r_j(x_0)$, which is not the maximum. Then, there exists a BS k that provides the maximum rate $r_k(x_0)$. Let ϕ_0^k be the value of the objective function when the UE at location x_0 is associated with BS k. The difference of the objective function due to the loads of BSs j and k is

$$\phi_0^k - \phi_0^* = \frac{\lambda(x_0)}{\mu(x_0)}\left(\frac{1}{r_k(x_0)} - \frac{1}{r_j(x_0)}\right) < 0, \tag{3.19}$$

which is a contradiction that ϕ_0^* is the minimum. Therefore, we conclude that at ϕ_0^* all the locations will be served by the BS that provide the highest rate. ■

3.7.2 Proof of Theorem 6

The proof is divided into the following two lemmas. The first lemma 11 gives the property that is required for proving the next lemma 12, which concludes the proof of the theorem 6.

Lemma 11 *Let $f_\alpha(r) = \sum_{i\in\mathscr{S}} \frac{r_i^{1-\alpha}}{1-\alpha}$. If $r \succ y$ then there is $A > 0$ large enough such that for all $\alpha \geq A$, $f_\alpha(r) > f_\alpha(y)$.*

Proof: Let $\alpha > 1$. Without loss of generality, assume that r and y are sorted in increasing order and that $r_1 > y_1$. Let $\delta = r_1 - y_1$. Then

$$(1-\alpha)(f_\alpha(r) - f_\alpha(y)) = \sum_{i=1\ldots n} \left(r_i^{1-\alpha} - y_i^{1-\alpha}\right) \tag{3.20}$$

$$\leq r_1^{1-\alpha} - y_1^{1-\alpha} + \sum_{i=2\ldots n} \left(r_i^{1-\alpha} - y_i^{1-\alpha}\right) \tag{3.21}$$

$$\leq r_1^{1-\alpha} - (r_1-\delta)^{1-\alpha} + (n-1)r_1^{1-\alpha} - \sum_{i=2\ldots n} y_i^{1-\alpha} \tag{3.22}$$

$$\leq n r_1^{1-\alpha} - (r_1-\delta)^{1-\alpha}. \tag{3.23}$$

Then we have $f_\alpha(r) > f_\alpha(y)$ if and only if

$$n r_1^{1-\alpha} - (r_1-\delta)^{1-\alpha} \leq 0 \tag{3.24}$$

$$\Leftrightarrow \frac{\log(n)}{\log\left(\frac{r_1-\delta}{r_1}\right)} \geq 1-\alpha \tag{3.25}$$

$$\Leftrightarrow 1 + \frac{\log(n)}{\log(r_1) - \log(r_1-\delta)} \leq \alpha. \tag{3.26}$$

■

Lemma 12 *Let $\mathscr{X}$ be a compact subset of $\mathbb{R}^{|\mathscr{S}|}$. Consider the set*

$$Z = \bigcap_{A>1} \overline{\bigcup_{\alpha\geq A} \underset{x\in\mathscr{X}}{argmax}\, f_\alpha(x)}.$$

Then Z is non-empty and is made of max-min vectors in $\mathscr{X}$.

Proof: Let $Z_A = \overline{\bigcup_{\alpha\geq A} \underset{x\in\mathscr{X}}{\operatorname{argmax}}\, f_\alpha(x)}$. It is a decreasing nested sequence of non-empty compact sets. By Cantor's intersection theorem, it is not empty and compact.

Let $x^* \in \bigcap_{A>1} Z_A$. There is an increasing sequence $\alpha(n)$ and $x_{\alpha(n)} \in \underset{x\in\mathscr{X}}{\operatorname{argmax}}\, f_{\alpha(n)}(x)$ with $x_{\alpha(n)} \to x^*$. Assume there is $y \succ x^*$. Then, by Lemma 11, there is N such that, for

every $n \geq N$, $f_{\alpha(n)}(y) > f_{\alpha(n)}(x^*)$. But, by definition of $x_{\alpha(n)}$, $f_{\alpha(n)}(x_{\alpha(n)}) \geq f_{\alpha(n)}(y)$, which is a contradiction with the fact that $f_{\alpha(n)}(x_{\alpha(n)}) \to f_{\alpha(n)}(x^*)$, which is ensured by Berge maximum theorem.

■

3.7.3 Proof of Proposition 7

Proof: We prove that the game Γ^{ϖ} is an ξ-potential game by verifying the definition 3 for the two BS types (macro or small). Consider action profiles $a = (a_i, a_{-i})$ and $b = (a'_i, a_{-i})$. First, consider a macro BS $i \in \mathscr{B}_e$. The neighbourhood of BS i includes all the BSs in the network, i.e., $N_i^{\varpi} = N_i^0 = \mathscr{S}$. With this, we have

$$U_i^{\varpi}(a) - U_i^{\varpi}(b) = \phi_{\alpha}(a) - \phi_{\alpha}(b).$$

Therefore, inequality in the definition 3 holds for all macro BSs.

Next, we show that it also holds for any small BS. Consider small BS $i \in \mathscr{B}_s$. The neighborhood of small BS i is given by (3.13). From the objective function (5.3) we have

$$\phi_{\alpha}(a) - \phi_{\alpha}(b) = \sum_{j \in \mathscr{S}} \frac{\big(1-\rho_j(a)\big)^{1-\alpha}}{\alpha - 1} - \sum_{j \in \mathscr{S}} \frac{\big(1-\rho_j(b)\big)^{1-\alpha}}{\alpha - 1}, \tag{3.27}$$

Since, the action of BS i only affects its neighbour BSs we have

$$\phi_{\alpha}(a) - \phi_{\alpha}(b) = \sum_{j \in N_i^0} \frac{\big(1-\rho_j(a)\big)^{1-\alpha}}{\alpha - 1} - \sum_{j \in N_i^0} \frac{\big(1-\rho_j(b)\big)^{1-\alpha}}{\alpha - 1}. \tag{3.28}$$

It can be splited as below.

$$\begin{aligned}\phi_{\alpha}(a) - \phi_{\alpha}(b) = &\sum_{j \in N_i^{\varpi}} \frac{\big(1-\rho_j(a)\big)^{1-\alpha}}{\alpha - 1} + \sum_{j \in N_i^0 \setminus N_i^{\varpi}} \frac{\big(1-\rho_j(a)\big)^{1-\alpha}}{\alpha - 1} \\ &- \sum_{j \in N_i^{\varpi}} \frac{\big(1-\rho_j(b)\big)^{1-\alpha}}{\alpha - 1} - \sum_{j \in N_i^0 \setminus N_i^{\varpi}} \frac{\big(1-\rho_j(b)\big)^{1-\alpha}}{\alpha - 1}.\end{aligned} \tag{3.29}$$

Let denote

$$\xi_i^{\varpi}(a,b) = \sum_{j \in N_i^0 \setminus N_i^{\varpi}} \frac{\big(1-\rho_j(a)\big)^{1-\alpha}}{\alpha - 1} - \sum_{j \in N_i^0 \setminus N_i^{\varpi}} \frac{\big(1-\rho_j(b)\big)^{1-\alpha}}{\alpha - 1}. \tag{3.30}$$

Then using (4.8) the above equation can be written as below.

$$\phi_\alpha(a) - \phi_\alpha(b) = U_i^{\varpi}(a) - U_i^{\varpi}(b) + \xi_i^{\varpi}(a,b). \tag{3.31}$$

Then we have

$$\left|U_i^{\varpi}(a) - U_i^{\varpi}(b) + \phi_\alpha(b) - \phi_\alpha(a)\right| \leq \max_{a,b\in X, i\in\mathscr{S}} \left|\xi_i^{\varpi}(a,b)\right|. \tag{3.32}$$

Hence, the inequality in the definition 3 also holds for all small BSs. Therefore, Γ^{ϖ} is an ξ-potential game with ξ given in Proposition 7. ■

3.7.4 Proof of Theorem 8

Proof: Let $a = (a_i, a_{-i})$ and $b = (a_i', a_{-i})$ be two action profiles, where BS i changes its action. Let $\rho_i(x,a) = \frac{\lambda(x)}{\mu(x)\nu_j(x,a)}$. Let denote $\nu_m = \max_{x,\bar{\theta},j\in\mathscr{S}} \frac{1}{\nu_j(x,\bar{\theta})}$, $\mu_m = \max_x \frac{1}{\mu(x)}$, and $\lambda_m = \max_{a\in X, j\in\mathscr{S}} \int_{x\in D_j(a)} \lambda(x)dx$.

The parameter ϖ is such that (3.13):

$$j \in N_i^0 \backslash N_i^{\varpi} \Rightarrow 0 < \frac{\int_x \lambda(x) 1_{i,j\in N_x} dx}{\int_x \lambda(x) 1_{i\in N_x} dx} < \varpi.$$

The change in load of BS j due to change in its strategy is as below:

$$\begin{aligned}\left|\rho_j(a) - \rho_j(b)\right| &= \left|\int_{x\in D_j(a_i,a_{-i})} \rho_i(x,a)dx - \int_{x\in D_j(a_i',a_{-i})} \rho_i(x,b)dx\right|, \\ &= \left|\int_{x\in D_j(a)\cap 1_{i,j\in N_x}} \rho_i(x,a)dx - \int_{x\in D_j(b)\cap 1_{i,j\in N_x}} \rho_i(x,b)dx\right|. \end{aligned} \tag{3.33}$$

The equation (3.33) is valid because the change of CRE bias of BS i affects the load of BS j only at those locations where both BSs i and j can potentially serve the users.

$$\left|\rho_j(a) - \rho_j(b)\right| \leq \int_{x\in 1_{i,j\in N_x}} \rho_i(x,a)dx + \int_{x\in 1_{i,j\in N_x}} \rho_i(x,b)dx. \tag{3.34}$$

$$\left|\rho_j(a) - \rho_j(b)\right| \leq \nu_m \int_{x\in 1_{i,j\in N_x}} \frac{\lambda(x)}{\mu(x)}dx + \nu_m \int_{x\in 1_{i,j\in N_x}} \frac{\lambda(x)}{\mu(x)}dx. \tag{3.35}$$

$$\left|\rho_j(a)-\rho_j(b)\right| \leq 2\mu_m \nu_m \int_{x\in 1_{i,j\in N_x}} \lambda(x)dx. \tag{3.36}$$

$$\leq 2\mu_m \nu_m \varpi \int_{x\in 1_{i\in N_x}} \lambda(x)dx, \tag{3.37}$$

$$\leq 2\mu_m \nu_m \lambda_m \varpi, \tag{3.38}$$

The inequality (3.37) is obtained by using (3.13).

Let $\rho_{\max}$ be the maximum load for a BS (assumed bounded away from 1). Consider function $g^\alpha : x \to \dfrac{(1-x)^{1-\alpha}}{\alpha-1}$. Its derivative is $(1-x)^{-\alpha}$. If $0 < x < \rho_{\max}$, then f is Lipschitz with constant $(1-\rho_{\max})^{-\alpha}$. This implies that for any x and y,

$$|g^\alpha(x) - g^\alpha(y)| \leq (1-\rho_{\max})^{-\alpha} |x-y|.$$

Then from (3.30) we have

$$\xi_i^\varpi(a,b) = \sum_{j\in N_i^0 \setminus N_i^\varpi} g_j^\alpha(a_i, a_{-i}) - g_j^\alpha(a_i', a_{-i}),$$

where $g_j^\alpha(a_i, a_{-i}) = \frac{\left(1-\rho_j(a_i,a_{-i})\right)^{1-\alpha}}{\alpha-1}$.

All in all, we obtain

$$|\xi_i(a,b)| = \left| \sum_{j\in N_i^0 \setminus N_i^\varpi} \frac{\left(1-\rho_j(a)\right)^{1-\alpha}}{\alpha-1} - \frac{\left(1-\rho_j(b)\right)^{1-\alpha}}{\alpha-1} \right|, \tag{3.39}$$

$$\leq \sum_{j\in N_i^0 \setminus N_i^\varpi} \left| \frac{\left(1-\rho_j(a)\right)^{1-\alpha}}{\alpha-1} - \frac{\left(1-\rho_j(b)\right)^{1-\alpha}}{\alpha-1} \right|, \tag{3.40}$$

$$\leq |\mathscr{S}| (1-\rho_{\max})^{-\alpha} \left|\rho_j(a)-\rho_j(b)\right|, \tag{3.41}$$

$$\leq |\mathscr{S}| (1-\rho_{\max})^{-\alpha} 2\mu_m \lambda_m \nu_m \varpi. \tag{3.42}$$

The constraint in Theorem 1 is satisfied if $|\xi| < \frac{\varepsilon}{2(|X|-1)}$. Using the above upper bound of ξ we get the sufficient condition in (3.17).

■

3.7.5 Neighborhood of macro BS

We show here that a similar neighborhood can defined for macro BSs. Let $\mathscr{L}_j$ be a set of all possible locations that can be served by BS j. It is defined as below:

$$\mathscr{L}_j = \{x \in \mathscr{L} : j \in N_x\},$$

where N_x is given by (3.12). We consider BS j as a neighbour to macro BS i if the load of BS j is significantly effected by the interference of macro BS i. Let denote a set of locations $J_i = \left\{x \in \mathscr{L} : \max_{\{\theta_i, f\}} P_i^f(\theta_i) g_i(x) \geq I^{\text{th}}\right\}$ and J_i' is its complement set. Formally, we define the neighborhood of macro BS i as:

$$N_i^{\varphi} = \left\{ j \in \mathscr{S} : \frac{\int_{x \in \mathscr{L}_j \cap J_i} \lambda(x) dx}{\int_{x \in \mathscr{L}_j} \lambda(x) dx} \geq \varphi \right\}. \tag{3.43}$$

According to the above equation, BS j is a neighbour of macro BS i if macro BS i interference exceeds a threshold I^{th} at more than φ proportion of locations that can be served by BS j. A similar idea for defining neighborhood based on interference is proposed in [24]. Note that the threshold I^{th} takes into account the effect of ABS on SINR in (5.2) and load in (3.6). Particularly, if $\max_{\{\theta_i, f\}} P_i^f(\theta_i) g_i(x) < I^{\text{th}}$ then the change in average data rate of BS j is bounded and we can come up with a similar condition for macro BSs as in Corollary 4 for the convergence of algorithms to optimal PNE.

Chapter 4

Learning Annealing Schedule of LLA

Annealing schedule describes the evolution of temperature parameter τ of Log-Linear Learning Algorithm (LLA). In this chapter, a new annealing learning algorithm (ALA) is proposed to learn the temperature parameter τ of LLA and Binary LLA (BLLA). ALA finds the optimal τ from an available set of τ by successively rejecting the worst performing τ. The process of successive reject in ALA gives a new annealing schedule. LLA overcomes the problem of choosing τ with the help of ALA. The performance of this new annealing schedule is compared with commonly used annealing schedules in the literature such as linear decreasing, log decreasing, and fixed parameter. It is observed from simulations that the new annealing schedule achieves lower global cost for a fixed time horizon compared to that of other annealing schedules. Two different versions of ALA are compared: first, ALA with linearly decreasing functions of τ and second, ALA with a fixed vector of τ. For lower time horizons, ALA with linearly decreasing functions of τ is better than ALA with a fixed vector of τ. Whereas, for higher time horizons, the performance of both the versions of ALA is same. Finally, we illustrate the application of ALA for the problem of load balancing among the base stations in heterogeneous networks considered in Chapter 3.

4.1 Introduction

Log-linear learning algorithm (LLA) was introduced in [12, 13] for learning a Nash Equilibrium (NE) of a potential game. Since then LLA has been extensive studied in the literature [11–20]. In LLA, a player chooses its action with a probability that depends on its utility, usually a larger probability for the actions with larger utility. A temperature τ parameter governs the perturbation of this probability. This perturbation allows LLA to

escape local minima [12, 13]. As $\tau \to \infty$ the perturbation increases resulting in a uniform probability of selection of actions. As $\tau \to 0$ the perturbation vanishes making the LLA behave as the best response algorithm.

LLA induces a reversible Markov chain on the action space of the game for a given τ. For potential games, the stationary distribution of the Markov chain concentrates on an NE as $\tau \to 0$. However, this convergence depends on the value of τ. For small τ the LLA may be stuck for a long time at a local minimum that corresponds to a suboptimal NE. Higher τ may result in unacceptable oscillations. A common practice in the literature is to empirically find a suitable temperature parameter τ. Usually, the value of temperature τ depends on specific applications. In general, it is not clear what is meant by small and higher τ with respect to an application. We address this problem of choosing automatically the parameter τ in this chapter.

We discuss few important papers from the related literature that characterize the convergence rate of LLA for a different class of potential games [18, 19, 73]. The result in [19] shows that the convergence time of LLA is of the order of exponential of the maximum drop where the maximum drop is defined as the maximum difference of two maxima of the potential function. In [18], the convergence time of LLA is exponential in the number of players of symmetric potential games. A modified log-linear learning algorithm whose convergence time is roughly linear in the number of players of a semi-anonymous potential game is presented in [73]. However, for general potential games, the effect of τ on the convergence properties is not well studied.

Another set of papers [26, 74] studies the annealing schedule of the temperature τ in similar settings. An annealing schedule describes the evolution of τ as a function of time. LLA converges in probability to the global minimum when $\tau(t) = \frac{\kappa}{\log(1+t)}$, where κ is the depth of the deepest local minimum of the potential function [26]. The κ represents an initial value of the temperature, which is generally unknown beforehand. Other heuristic annealing schedules such as linear decreasing τ and fixed τ are used in the literature [74]. However, if the initial τ is very high then these annealing schedules also take a long time for convergence, which may not be suitable for many applications. It is not clear how to choose the initial temperature τ of an annealing schedule so as to guarantee the desirable convergence. This is the focus of this chapter, to develop an annealing schedule that is fast and guarantees asymptotic convergence.

We propose a new distributed annealing learning algorithm (ALA) to learn the annealing schedule of the parameter of LLA. ALA is developed by adapting successive reject

algorithm (SRA) [2]. In ALA, a time horizon is considered and a set of values of τ of wide range is considered to run the LLA. In the evolution of LLA, ALA checks the performance of different τ to successively reject the ones with the worst performance. At the end of the horizon, ALA selects the best performing τ. The process of successively rejecting the set of bad τ results into an annealing schedule for LLA. We will see that this annealing schedule is fast, performs better than other annealing schedules, and guarantees asymptotic convergence. To evaluate the performance and to show an application of the proposed ALA we apply ALA for load balancing problem in heterogeneous networks considered in Chapter 3. We use only Cell Range Extension (CRE) bias association for load balancing. We skip Almost Blank Subframe (ABS) technique in order to simplify the system model.

The rest of the chapter is organized as follows. We first describe a simplified heterogeneous network model in Section 4.2. Then we described ALA in Section 4.3. In Section 4.4, the performance of ALA is studied using extensive simulations. Finally, the conclusions are given in Section 4.5.

4.2 System Model

We consider a simplified heterogeneous cellular system model of Chapter 3 to study the annealing schedule of LLA. In the following, we describe the important aspects of it for the sake of completeness. A downlink user association scenario of the cellular network is considered. The sets $\mathscr{B}_e$ and $\mathscr{B}_s$ denote the set of macro BSs and set of small BSs, respectively. The set of all stations is denoted $\mathscr{S} \triangleq \mathscr{B}_e \cup \mathscr{B}_s$. Every BS i maintains a CRE bias parameter $c_i \in \{1, \ldots, c_{\max}\}$. The CRE bias vector is denoted $\bar{\boldsymbol{c}} = \left[c_1, c_2, \ldots, c_{|\mathscr{S}|}\right]$. A model scenario is shown in Fig. 4.8, where BS 1 is a macro BS and others are small BSs. The coverage regions of BSs are shown with different colors. The irregular coverage regions of BSs are due to the fading channel model, which is described below.

Channel Model

We consider a simplified version of channel model of Chapter 3. Since we do not consider ABS the expression of Signal-to-Noise Ratio (SINR) becomes simple, which is described as follows. The received power at location x from BS i is $P_i g_i(x)$, where P_i is the transmit power and $g_i(x)$ is the channel gain, which captures the effect of path-loss, auto-correlated and cross-correlated shadowing. The effect of small-scale fading is not considered because the time for user association procedure is assumed to be much larger than the channel

coherence time [37]. We consider a scenario where the locations of the BSs and of the users during their download are fixed. Therefore, the shadow fading component is a constant multiplicative factor, which does not change with time. Formally, the channel gain model considered is [59]:

$$g_i(x) = \min\left\{1, K\left|x - x_i\right|^{-\eta} e^{\beta y_i(x)}\right\}, \tag{4.1}$$

where $K = \left(\frac{\lambda_w}{4\pi d_0}\right)^2$, λ_w is the wavelength, d_0 is the reference distance, x_i is the location of the BS i, $\eta \geq 2$ is the path-loss exponent, and $e^{\beta y_i(x)}$ is the shadowing component where $\beta = \frac{\log 10}{10}$ and $y_i(x)$ is a realisation of Gaussian random process of zero mean and covariance function $C_{y_i}(\Delta x)$ [60]:

$$C_{y_i}(\Delta x) = \sigma_{sh}^2 e^{-\frac{\Delta x}{D_c}}, \tag{4.2}$$

where σ_{sh}^2 is the variance, Δx is the displacement, and D_c is the decorrelation distance [59]. A constant cross correlation between the $y_i(x)$ and $y_j(x)$ is considered as in [61].

The SINR $\gamma_i(x)$ of a user at location x is given as:

$$\gamma_i(x) = \frac{P_i g_i(x)}{\sum_{j \in \mathscr{S}} P_j g_j(x) + N_0}, \tag{4.3}$$

where $N_0 = -174 + 10\log W$ is thermal noise power in dBm and W is system bandwidth in Hz.

CRE User Association Rule

According to this rule, a user located at x is associated to the BS i that provides the highest biased received power. The set of locations $\mathscr{D}_i(\bar{\boldsymbol{c}})$ associated to BS i is defined as:

$$\mathscr{D}_i(\bar{\boldsymbol{c}}) = \left\{x | \forall j \in \mathscr{S}, P_i g_i(x) c_i \geq P_j g_j(x) c_j\right\}. \tag{4.4}$$

This kind of user association rule based on biased received power is commonly used in the heterogeneous networks [31, 43–45, 62–68].

Traffic Model

BSs are modeled as M/G/1/PS queues as in Chapter 3. The users at location x arrive with a rate $\lambda(x)$ [arrivals/s/m^2] and with an average demand of $1/\mu(x)$ [bits]. The load of BS i is

given below[1].

$$\rho_i(\bar{\boldsymbol{c}}) = \int_{\mathscr{D}_i(\bar{\boldsymbol{c}})} \frac{\lambda(x)}{\mu(x)\nu_i(x)} 1_{\{\gamma_i(x) \geq \gamma_{\min}\}} dx, \tag{4.5}$$

where the data rate provided is $\nu_i(x)$ [bits/s]. A user is served only if its SINR exceeds a minimum threshold $\gamma_{\min}$. BS i is stable if and only if $0 \leq \rho_i < 1$. In this work, only stable network configurations are considered. The load vector is denoted as $\boldsymbol{\rho} = \left[\rho_1, \rho_2, \ldots, \rho_{|\mathscr{S}|}\right]$.

4.2.1 Problem Formulation and Objective Function

As in Chapter 3 the goal is to minimise an $\alpha-$fairness function $\phi_\alpha(\bar{\boldsymbol{c}})$ over a feasible set $\mathscr{F}$. Since ABS is not considered the $\alpha-$fairness function $\phi_\alpha(\bar{\boldsymbol{c}})$ and the feasible set $\mathscr{F}$ become simple, which are defined as below:

$$\phi_\alpha(\bar{\boldsymbol{c}}) = \begin{cases} \sum_{i\in\mathscr{S}} \frac{(1-\rho_i(\bar{\boldsymbol{c}}))^{1-\alpha}}{\alpha-1}, & \alpha \geq 0, \alpha \neq 1, \\ -\sum_{i\in\mathscr{S}} \log\left(1-\rho_i(\bar{\boldsymbol{c}})\right), & \alpha = 1, \end{cases} \tag{4.6}$$

$$\mathscr{F} = \{\bar{\boldsymbol{c}} | 0 \leq \rho_i(\bar{\boldsymbol{c}}) < 1, \forall i \in \mathscr{S}\}. \tag{4.7}$$

The function $\phi_\alpha(\bar{\boldsymbol{c}})$ captures various aspects of fairness and performance for the network depending on the choice of α. In particular, it results into a rate-optimal policy for $\alpha = 0$, proportional fairness for $\alpha = 1$, average delay for $\alpha = 2$, and min-max load policy as $\alpha \to \infty$ [1,37,69,75].

4.3 Learning Annealing Schedule of LLA

In this section, we first present a potential game model for user association and then describe LLA. Next, we describe the learning algorithm for annealing schedule of LLA.

Definition 13 *[User Association Game] It is defined by the tuple* $\Gamma = \left\{\mathscr{S}, \{X_i\}_{i\in\mathscr{S}}, \{U_i\}_{i\in\mathscr{S}}\right\}$, *$\mathscr{S}$ is a set of BSs, X_i is a set of strategies of BS i. Strategy set X_i is a discrete set of CRE bias values. Let denote a_i and a_{-i} as the strategy of player i and strategies of players except i, respectively. The cost function U_i is given below:*

$$U_i(a_i, a_{-i}) = \sum_{j\in N_i^0} \frac{(1-\rho_j(a_i, a_{-i}))^{1-\alpha}}{\alpha-1}, \tag{4.8}$$

[1]Note that the expression of load of BS in (4.5) is simplified version of (3.6) as ABS is not considered.

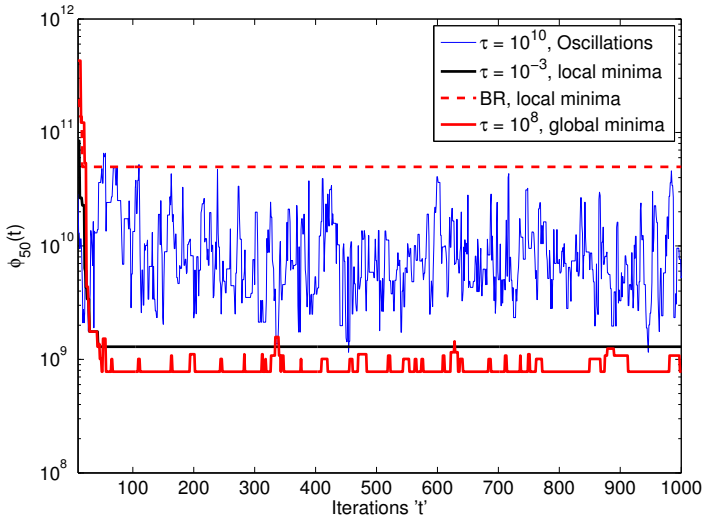


Figure 4.1: Illustration of effect of τ on LLA. High value of $\tau = 10^{10}$ results in oscillations of LLA. While, for the small value of $\tau = 10^{-3}$ LLA gets stuck into a local minimum. Best Response (BR) also gets stuck into a local minimum. A carefully chosen value of $\tau = 10^{8}$ makes the LLA reach the global minimum.

where $\rho_j(a_i, a_{-i})$ is the load of BS j given in (4.5) and N_i^0 is the neighborhood of BS i that is obtained by substituting the neighborhood control parameter $\varpi = 0$ in (3.13).

The above game Γ is an exact potential game with the potential function ϕ_α in (5.3). An exact potential game has, at least, one pure Nash equilibrium (PNE) and local optimizers of the potential function are PNEs [3]. A PNE of a game is reached when no player can benefit by changing its strategy unilaterally.

LLA and BLLA algorithms described in Chapter 3 guarantee the asymptotic convergence to the optimal NE of the exact potential game Γ [20, 75]. This means that, asymptotically, the probability that LLA is at the global minimum approaches to one as τ goes to zero. The effects of τ parameter on LLA is shown using simulations in Fig. 4.1. A fixed $\alpha = 50$ is considered. For high values of $\tau = 10^{10}$, LLA results into oscillations. This is due to the fact that the algorithms converge fast in probability to the uniform distribution. As a matter of fact, it doesn't spend much time in optimal states, which is not practically desirable. For small values of τ, asymptotically, the algorithms will spend most of the time in the global optimal. However, convergence is slow in probability. This explains that the system can take a long time to escape from sub-optimal states. The best response (BR) algorithms get trapped into local minima. Contrary to best response, however, the LLA does not get stuck in sub-optimal states.

4.3.1 Annealing Learning Algorithm for LLA

Algorithm 6 Annealing Learning Algorithm

1: **Initialisation:** Let T be a time horizon and M be the number of τ parameters. $A_1 = \{\tau_1, \ldots, \tau_M\}$ and $\Upsilon = \frac{1}{2} + \sum_{i=2}^{M} \frac{1}{i}$.
2: **for** Phase $k = 1, 2, \ldots, M-1$ **do**
3:

$$n_k = \left\lceil \frac{1}{\Upsilon} \frac{T-M}{M+1-k} \right\rceil \tag{4.9}$$

and $n_0 = 0$.
4: **for** each $i \in A_k$ **do**
5: play LLA with τ_i for $n_k - n_{k-1}$ iterations.
6: **end for**
7: Compute

$$A_{k+1} = A_k \backslash arg\,max_{i \in A_k} \hat{X}_{i,n_k}, \tag{4.10}$$

where $\hat{X}_{i,n_k} = \frac{1}{n_k} \sum_{t=1}^{n_k} X_{i,t}$, where $X_{i,t}$ is the cost obtained using (4.8) with τ_i at time t.
8: **end for**

An annealing schedule with quick convergence with some guarantee is desirable. We propose to learn the annealing schedule for a fixed time horizon. The idea is to explore the given values of τ and reject a τ value whose average cost is the worst compared to other τ values. At the end of the time horizon, the surviving τ value is the best value to run LLA.

ALA is a successive reject algorithm originally proposed for the best arm identification of multi-arm bandit problem [2]. We adapt SRA for LLA to identify the best τ. The multiple τ parameters are multiple arms of SRA. Every BS runs this algorithm distributively to obtain its own best τ to run LLA.

The details of ALA are described in Algorithm 6. Consider a M number of τ parameters of a wide range. First, the algorithm divides the given time horizon (i.e., the T iterations) in $M-1$ phases. At the end of each phase, the algorithm dismisses the τ_i parameter with the highest empirical mean cost, $\hat{X}_{i,s} = \frac{1}{s} \sum_{t=1}^{s} X_{i,t}$, where $X_{i,t}$ is the cost obtained using (4.8) with τ_i at time t. During the next phase, it plays equally often each τ which has not been dismissed yet. The recommended τ is the last surviving τ.

The length of the phases is carefully chosen to obtain an optimal convergence rate for finite horizon as in multi-arm bandit setting [2]. The following theorem by [2] gives the probability of error of finding the best τ. Consider a stochastic multi-armed bandit setting with M arms generating stochastic rewards according probability distribution $\nu_i, i = 1, 2, \ldots, M$ with average rewards $\mu_i, i = 1, 2, \ldots, M$. Let μ^* be the reward of the best arm

Table 4.1: Simulation parameters.

Parameter	Variable	Value
Number of BSs	N_s	8
Transmit power of macro BS	P_{macro}	46 dBm
Transmit power of small BS	P_{small}	24 dBm
Average file size	$\frac{1}{\mu}$	0.5 Mbytes
Average traffic load density	$\frac{\lambda}{\mu}$	64 bits$/s/m^2$
System bandwidth	W	20 MHz
Carrier frequency	f_c	2.6 GHz
Noise power	N	-174+10log(W) dBm
Minimum SINR	γ_{min}	-10 dB
Path-loss exponent	η	3.5
Reference distance	d_0	10 m
CRE bias set	c_i	$\{1,2,\ldots,16\}$

and let $H_2 = \max_{i=1,2,\ldots,M} \frac{i}{(\mu^* - \mu_i)^2}$.

Theorem 9 *The probability of error $Pr(T)$ of ALA satisfies*

$$Pr(T) \leq \frac{M(M-1)}{2} \exp - \left(\frac{T-M}{\Upsilon H_2} \right). \tag{4.11}$$

The above result is applicable in our setting since each LLA with a τ parameter acts an arm of a stochastic multi-armed bandit. LLA generates potential function values according to a probability distribution. These potential functions values are observed as rewards. Note that ALA also guarantees asymptotic convergence to the global optimal of the user association game because at the end of time horizon ALA runs LLA with a single best τ that guarantees asymptotic convergence to global optimal.

4.4 Simulation Results

In this section, we evaluate the performance of ALA considering the standard parameters as adopted in 3GPP [71]. These parameters are listed in the Table 4.1. We consider 8 BSs located in a two-dimensional region $\mathscr{L}$. BS 1 is a macro BS that transmits with P_{macro} and the rest are small BSs that transmit with P_{small}. The user traffic varies with location across an average traffic density of 64 bits$/s/m^2$. There are two hotspots where the traffic is 5 times the average traffic. We consider shadow fading with a standard deviation of $\sigma_{sh} = 8$ dB and a decorrelation distance of $D_c = 20$ m. Cross-correlation between the

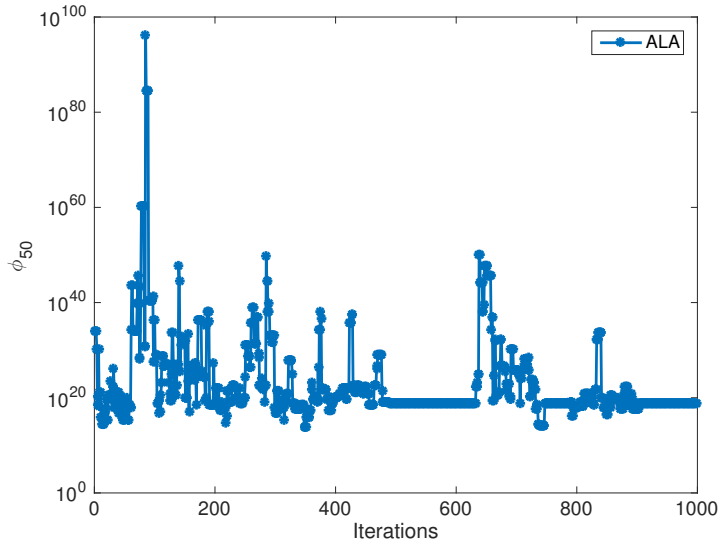


Figure 4.2: Evolution of LLA along with ALA with $\boldsymbol{\tau} = \left[10^{12}, 10^{10}, 10^{8}, 10^{2}\right]$. In the first few iterations there are lot of fluctuations since ALA explores all the values of τ. As ALA rejects the worst performing τ values the fluctuations decreases. The best performing parameter $\tau = 10^{8}$ is chosen by ALA at the end of time horizon.

shadowing components at a location is considered to be 0.5. We use the following classical Shannon formula for calculating channel capacity $\nu_i(x)$ at any location x:

$$\nu_i(x) = W \log_2 \left(1 + \gamma_i(x)\right). \tag{4.12}$$

The performance of ALA with $\tau = \left[10^{12}, 10^{10}, 10^{8}, 10^{2}\right]$ is shown in Fig. 4.2 for $\alpha = 50$. In the first few iterations, a lot of oscillations are observed. This is because of the high values of τ used by ALA to explore through all the given τ parameters. As the algorithm progresses higher values of τ are rejected and the algorithm progresses towards stabilization. At the end, the ALA runs LLA with the surviving $\tau = 10^{8}$ and it converges. We see that the smallest $\tau = 10^{2}$ is not always the best parameter since LLA can get stuck into local minimum with small τ. The evolution of potential function obtained from LLA with the given smallest $\tau = 10^{2}$ is shown in Fig. 4.3. LLA converges fast to some local minima and gets stuck into it most of the time. Since, the time horizon is finite the behaviour of LLA with small τ is same as BR because it does not have enough time to escape from the local minimum.

The performance of a linear decreasing annealing schedule is shown in Fig. 4.4. Note that the initial value of τ is the highest value of the given set of τ parameters. It performs

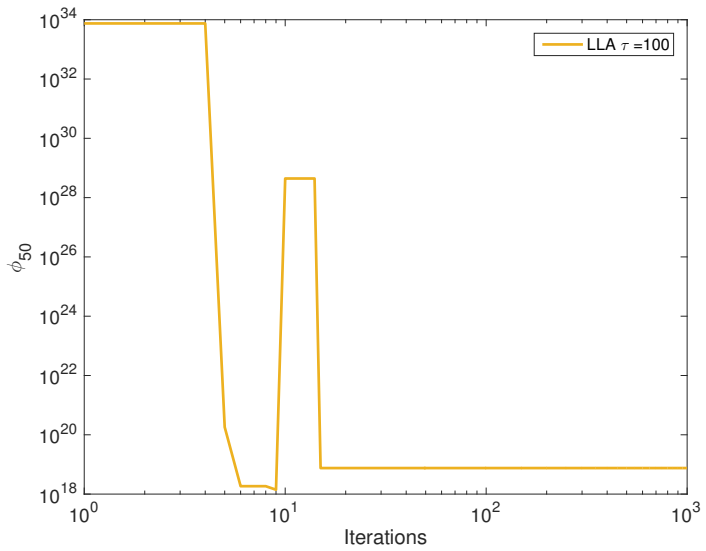


Figure 4.3: Evolution of LLA with the available smallest $\tau = 10^2$. LLA converges fast to some local minimum similar to as BR algorithm.

poor and results in a lot of oscillations that is not desirable. It shows that linear annealing schedule may not be suitable for load balancing. We arrived at the same conclusion with the performance of log-decreasing annealing schedule shown in Fig. 4.5.

Now, we study the performance of two different versions of ALA with different functions of τ. First version of ALA with fixed $\boldsymbol{\tau} \in \left[10^{12}, 10^{10}, 10^{8}, 10^{2}\right]$ is A1 and second version of ALA with linearly decreasing functions $\boldsymbol{\tau} \in \left[10^{12}/t, 10^{10}/t, 10^{8}/t, 10^{2}/t\right]$ is A2. The probability of improvement is defined as the probability of a better value of the potential function obtained using an algorithm compared to that of using another algorithm. The probability of improvement of A1 and A2 compared to various annealing schedules is shown in Fig. 4.6. This probability is computed by averaging over 1000 realizations. It is observed that improvement probability of A1 and A2 compared to smallest $\tau = 100$ increases quickly as the number of iterations increases and reaches unity at 400 iterations. This shows both A1 and A2 are certainly better than LLA with the smallest τ after 400 iterations. However, A2 is faster than A1. Both A1 and A2 are better than LLA with linear and log annealing schedules. Particularly, they are better than linear and log annealing schedule more than 70% and 60% of times, respectively.

The performance comparison of A1 and A2 is shown in Fig. 4.7 for $\alpha = 50$. The probability curves in the figure are computed by averaging over 500 realizations. The probability of A1 being strictly better than A2 is small in the beginning, decreases with

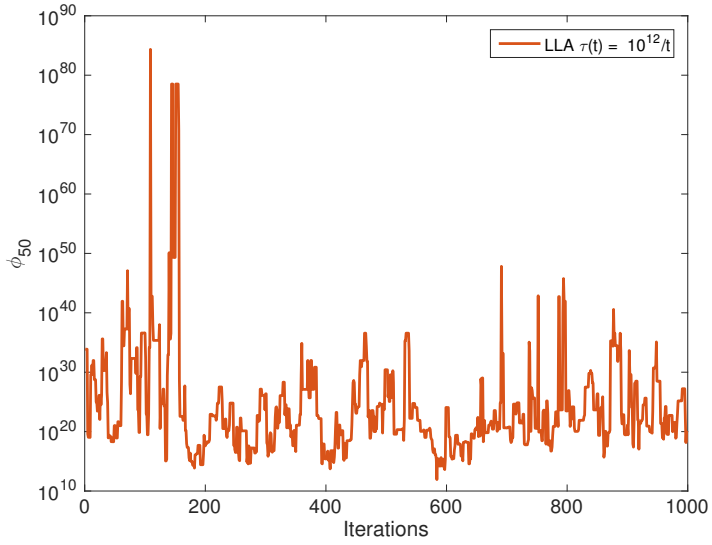


Figure 4.4: Evolution of LLA with linear decreasing $\tau(t) = 10^{12}/t$ with time t. As the initial value of τ is large LLA results into oscillations and takes a long time to stabilize. For a fixed horizon this behaviour is undesirable making this annealing schedule impractical.

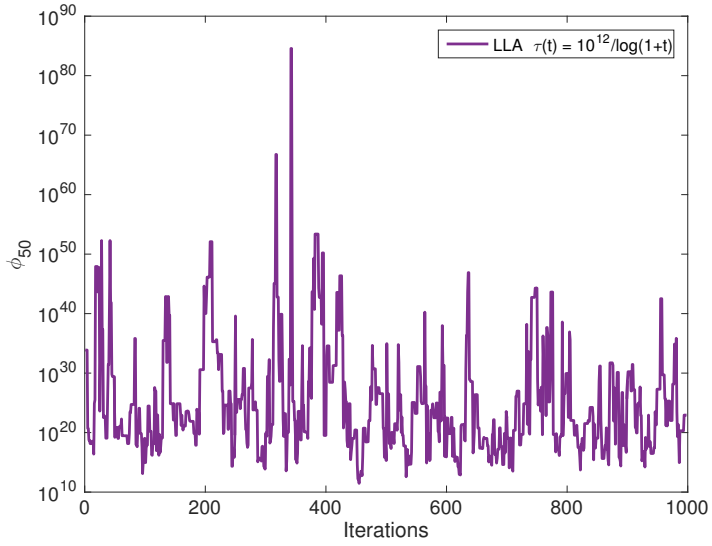


Figure 4.5: Evolution of LLA with log decreasing $\tau(t) = 10^{12}/\log(1+t)$. This annealing schedule is much slower than the linear decreasing annealing and hence a lot of oscillations are observed.

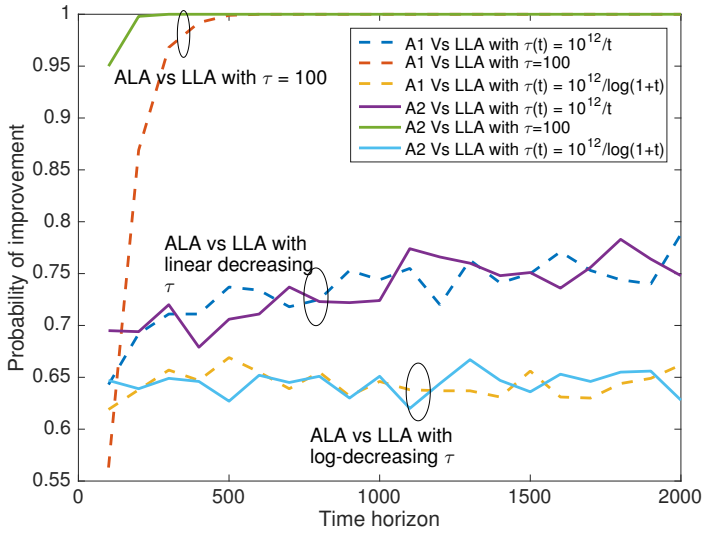


Figure 4.6: Performance comparison of ALA A1 with fixed $\boldsymbol{\tau} \in \left[10^{12}, 10^{10}, 10^{8}, 10^{2}\right]$ and ALA A2 with linearly decreasing $\boldsymbol{\tau} \in \left[10^{12}/t, 10^{10}/t, 10^{8}/t, 10^{2}/t\right]$. This figure shows that both A1 and A2 are certainly better than LLA with the smallest $\tau = 100$ after 400 iterations.

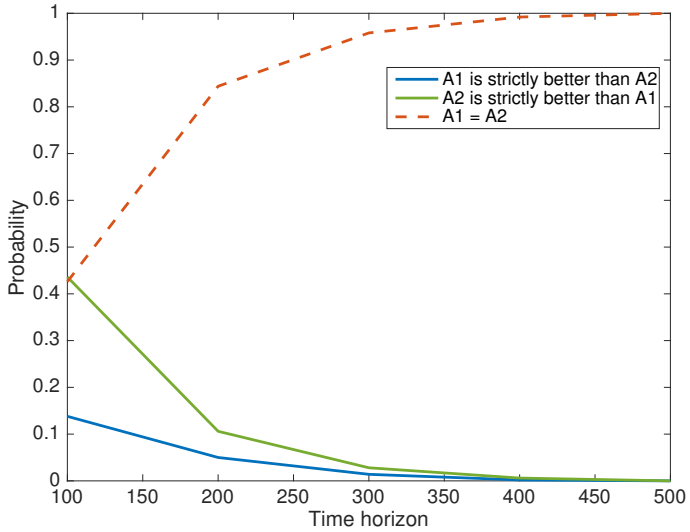


Figure 4.7: Comparison of algorithms A1 and A2. For low time horizons, it is better to use A2 as it has higher probability of improvement compared to that of A1. For higher time horizons, either A1 or A2 can be used as the probability of improvement is same for both.

Table 4.2: In this table, we comparison of optimal loads obtained using LLA along with ALA for $\alpha = 0$ and $\alpha = 50$. As in Chapter 3 we see that the loads of the BSs are balanced for $\alpha = 50$.

	$\alpha = 0$		$\alpha = 50$	
BS i	c_i	$\rho_i\%$	c_i	$\rho_i\%$
1	1	92	1	45
2	1	7	8	42
3	1	4	9	23
4	1	9	8	37
5	1	11	7	37
6	1	8	7	43
7	1	5	8	30
8	1	7	6	37

time, and becomes zero at 400. Whereas, the probability of A2 being strictly better than A1 is higher in the beginning and then decreases to zero. The probability of A1 equal to A2 is higher, in the beginning, increases with time, and becomes unity at 400. Therefore, for low time horizons, it is better to use A2. For higher time horizons, A1 or A2 can be used.

We now compare in Table 4.2 the optimal BS loads obtained using algorithm A1 for different α. As in Chapter 3 we see that the loads of the BSs are balanced for $\alpha = 50$ using LLA with ALA. As said earlier, the case with $\alpha = 0$ gives rate optimal policy, which is obtained when all biases equal to unity. This is verified by the output of ALA in Table 4.2. This corresponds to the classical best signal association rule that results in heavy load imbalance between stations: the load of the macro BS reaches 92%, while small BSs have loads less than 11%. Min-max policy is obtained with a value of $\alpha = 50$. A min-max load vector of BSs is obtained as shown in Table 4.2. Note that the load of macro BS is reduced to 45%. The corresponding coverage regions with optimal CRE bias for $\alpha = 50$ are shown in Fig. 4.8.

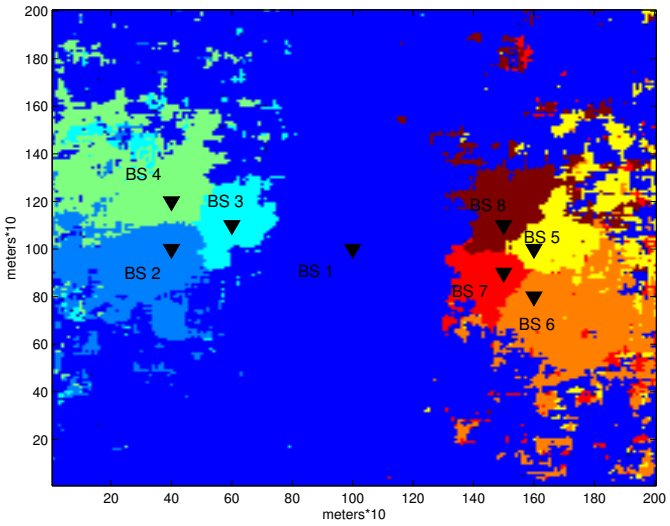


Figure 4.8: Optimal coverage regions obtained using optimal CRE biases for $\alpha = 50$ under correlated shadow fading.

4.5 Conclusions

In this Chapter, a new distributed annealing learning algorithm for learning the annealing schedule of LLA over a fixed time horizon is proposed. The LLA with the new annealing schedule is certainly better than with the fixed τ parameter. Also, the improvement probability of the proposed annealing schedule increases with the iterations when compared to linear decreasing and log decreasing annealing schedules of LLA. For lower time horizons, ALA with linearly decreasing τ is better than ALA with a fixed vector of τ. Whereas, for higher time horizons, ALA performance is same in both the cases.

Chapter 5

Channel Assignment in D2D Wireless Networks using Distributed Learning in Noisy-Potential Game

We present an application of the developed Binary Log-linear Learning Algorithm (BLLA) for noisy potential games. We apply BLLA for Channel Assignment Problem (CAP) in Device-to-Device (D2D) wireless networks. CAP is known to be NP-hard in the literature and there is no practical learning algorithm that is optimal and that takes into account the estimation noise. In this chapter, we first formulate the CAP as a stochastic optimization problem to maximize the expected sum data rate. Then this optimization problem is turned into a noisy potential game. The throughput estimation noise of the users is captured using noisy potential games. Then, we apply the developed distributed BLLA that converges to the optimal channel assignments. We assess the performance of BLLA by extensive simulations. It shows that the sum data rate increases with the number of channels and users. Contrary to the better response algorithm, the proposed BLLA achieves the optimal channel assignments distributively even in presence of estimation noise.

5.1 Introduction

Ever increasing demand for higher data rates of mobile users and scarcity of wireless frequency spectrum is making efficient utilization spectrum resources increasingly critical. Device-to-Device (D2D) networks increase the utilization of the spectrum resources by providing spatial spectrum reuse [76–87]. Moreover, D2D networks provide offloading

of cellular network, reduction of communication costs and computation powers among the devices [77], provide a backup or secondary network in case of damage to the cellular network during natural disaster, provide many applications like event video stream sharing and context-aware services [77]. A D2D network consists of Base Stations (BSs), cellular User Equipments (UEs), and D2D UEs. UEs can either be cellular or D2D depending on the mode selection. Cellular UEs communicate on license bands with a BS of the coverage cell while the D2D UEs reuse the licensed bands without or with the limited help of BSs. However, channel reuse creates co-channel interference that degrades the quality of service (QoS) of the cellular UEs. Therefore, a crucial problem in underlay D2D networks is to assign channels to UEs so that the total interference per channel is low.

Channel Assignment Problem (CAP) in D2D networks is challenging due to the lack of perfect Channel State Information (CSI) at BS. Estimated CSI of cellular UEs can be feedback to BS but the CSI of D2D links is difficult to obtain. However, the estimation noise effects the performance. The estimation noise can arise due to several factors such as randomly varying channel gain, feedback errors, feedback delay errors, and quantization errors [59]. Moreover, in a centralized network, the feedback overhead becomes infeasible. Therefore, it is essential to have a low feedback distributed solution that takes into account the CSI estimation noise, and that achieves optimal channel assignment. Focusing on these requirements we present the main contributions of this chapter.

5.1.1 Contributions

A summary of the main contributions of this chapter is as follows.

- *Novel Approach:* Our approach is to learn the optimal channel assignments in a D2D wireless network using a noisy potential game that takes into account the estimation noise. Distributed learning in a noisy environment for CAP is novel. We consider a SOP with the objective to maximize the expected sum data rate of an underlay D2D network. We translate this problem into a noisy potential game. The notion of the noisy potential game is introduced to account for the fact that only noisy estimates of the utility are available to the players.

- *Learning algorithm:* We apply distributed BLLA for the CAP in D2D networks. BLLA solves CAP in D2D networks to achieve the optimal channel assignments that correspond to optimal Nash equilibrium of the game.

- *Simulations results:* Extensive simulations show that BLLA achieves the maximum sum data rate of the network. It shows that BLLA works for both downlink and uplink. It shows that BLLA well tracks the increase of sum data rate with the number of UEs and with the number of channels. We also show that contrary to the better response algorithm, BLLA converges to the optimum even in presence of estimation noise.

5.1.2 Related literature survey and comparison

In this subsection, we discuss and compare different approaches for CAP in the literature. CAP in wireless networks is a standard problem and it is known to be NP-hard [88]. Extensive surveys of CAP can be found for in underlay D2D networks in [76], for Orthogonal Frequency Division Multiple Access (OFDMA) systems in [89], for heterogeneous networks in [90], and for cognitive radio networks [89].

The CAP solution approaches adopted by the state-of-the-art channel assignment algorithms are dynamic programming [85], graph-theoretical and heuristic solutions [86,87], game theory [78–84], linear programming (LP), non-linear programming (NLP), and Markov Random Field [89]. Other approaches for CAP are neural networks [91], simulated annealing [92], tabu search, genetic algorithms [90]. We discuss closely related works in D2D in the following.

In [85], the authors jointly optimize the mode selection and channel assignment in a cellular network with underlying D2D communications in order to maximize the weighted sum rate. A dynamic programming (DP) algorithm is proposed but it is exponentially complex. Therefore, a suboptimal greedy algorithm is proposed. However, this approach relies on explicit closed form expressions of sum data rate for different channel fading scenarios. In contrast, our approach does not need closed form expressions of the sum data rate. It can be applied for any fading scenario as it just needs the measured throughput of few users on only two channels in any iteration.

In [86], a suboptimal graph-theoretical heuristic solution for CAP in D2D networks is proposed. The weighted signal sum is maximized using maximum-weighted bipartite matching and interference sum is minimized using minimum-weighted partitioning. This approach is centralized since the BS uses the partial CSI of all the UEs. Whereas, our approach is distributed, optimal, and it maximizes the sum throughput instead of a signal sum or interference sum.

In [87], a heuristic algorithm is proposed for joint mode selection, channel allocation

and power allocation in a D2D wireless network. The channel gain of the different links measured through pilot signal is considered to be exact without noise. Whereas, our approach takes into account the effect of estimation noise of throughput.

Different game-theoretic models such as non-cooperative games, coalition formation games, and auction games are used to study the radio resource allocation issues in D2D networks [78–84]. In [84], a game-theoretical reverse iterative combinatorial auction is proposed as the allocation mechanism to optimize the system sum rate over the resource sharing of both D2D and cellular modes of the users. In this auction model, the different channels compete for the D2D links. The auction has proved to be cheat proved that means channels allocated to links with the highest bid. However, in this auction, the BS needs to have the bid from all the D2D links that create a huge feedback overhead.

In [79], a pricing mechanism is proposed to maximize the network throughput under QoS constraint. The base stations transmit a pricing signal to the D2D users. The price increases with the interference from the D2D links. The D2D devices utilize the pricing signal to update their strategies to maximize their data rates. However, the algorithm proposed is a heuristic algorithm whose performance is only evaluated through simulations. In contrast, BLLA's convergence is proven theoretically and confirmed through simulations.

In [80], the uplink resource allocation problem for multiple D2D and cellular users is modeled as a coalition game. The utility function combines different modes, mutual interferences, and resource sharing policy. Convergence to a Nash equilibrium is proved. However, the equilibrium may be sub-optimal and inefficient.

In order to improve the energy efficiency of wireless users, the joint mode selection and spectrum sharing as a coalition formation game is modeled and a coalition formation algorithm is proposed to jointly solve the mode selection and spectrum sharing in a D2D system in [81]. The algorithm converges to a stable coalition. However, a stable coalition may not be optimal. Whereas, BLLA is proven to be optimal.

In [82], the joint mode selection and spectrum sharing problem are modeled as a coalition formation game. A distributed coalition formation algorithm is proposed and its performance is evaluated through simulations. In [83], a contract-based game theoretic mechanism to resolve the unknown channel quality problem by eliminating the incentive of users to report untruthfully with designed service contracts is proposed. Its performance is evaluated through simulations only.

The rest of the chapter is organized as follows. The system model and problem formulation are described in Section 5.2. A noisy potential game model for D2D network is

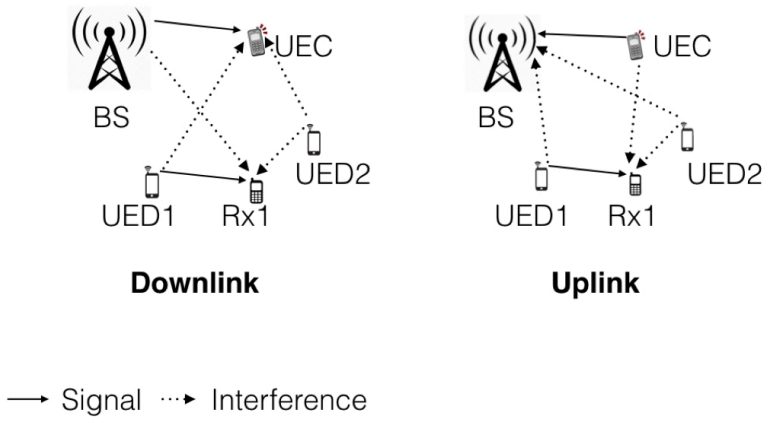


Figure 5.1: [D2D cellular network model]: This figure shows the signal and interference links in downlink and uplink. In the downlink, D2D users UED1 and UED2 causes co-channel interference at cellular user UEC. Also, the BS and UED2 causes co-channel interference at the RX1. Whereas, in the uplink, UED1 and UED2 cause interference at BS; and UEC and UED2 cause interference at RX1.

described in Section 5.3. The adapted BLLA for D2D networks is described in Section 5.4. Simulation results and conclusions are presented in Sections 5.5 and 5.6, respectively.

5.2 D2D Cellular Network Model

In this section, we first describe the D2D cellular network model that is shown in Fig. 5.1. Then, we describe the channel model considered in this chapter. Finally, we formalize the problem statement.

Figure 5.1 shows downlink (DL) and uplinks (UL) of a D2D network model. We consider a single base station (BS) and two types of UEs: (i) cellular UEs (UECs) that communicate with the BS and (ii) D2D UEs (UEDs) that communicate with other UEDs. All UEs are randomly located in the region. The set of UEs is denoted as $\mathscr{D}$. We consider a set of orthogonal frequency channel bands $\mathscr{F}$. The UECs are assigned different channels by the BS, whereas UEDs reuse these channels. A UE transmits on a single channel. The UEs that transmit on the same channel $c \in \mathscr{F}$ cause co-channel interference to each other. In DL, BS transmits a signal to UEC while causing interference at the receivers of UEDs that use the same channel as shown in Fig. 5.1. A UED transmits a signal to its receiver

(Rx) while causing interference to the UECs and receivers of other UEDs that are on the same channel. In UL, UEC transmits a signal to the BS while causing interference to UEDs, see Fig. 5.1. Also, UEDs cause interference to the BS and other UEDs in UL. The amount of interference caused depends on the channel gain between the devices. The channel model is described below.

5.2.1 Channel Model

We consider a channel model that captures the effect of pathloss, shadowing, and small-scale fading. Formally, the channel model considered is [59]:

$$g(d) = \min\left\{1, K\,|d|^{-\eta}\, e^{y}\hbar\right\}, \tag{5.1}$$

where $K = \left(\frac{\lambda_w}{4\pi d_0}\right)^2$, λ_w is the wavelength, d_0 is the reference distance, d is the distance between the receiver and transmitter, $\eta \geq 2$ is the pathloss exponent, and $e^{\beta y}$ is the shadow fading component where $\beta = \frac{\log 10}{10}$ and y is a Gaussian random variable of zero mean and σ_{sh}^2 variance. The small scale fading component is $\hbar$. We consider Rayleigh fading model. Therefore, $\hbar$ is a unit mean exponential random variable.

Let denote $\mathscr{D}(c)$ as the set of UEs on channel $c \in \mathscr{F}$. Let P_i and N denote the transmit power of UE i and noise power, respectively. The signal-to-interference-plus-noise ratio (SINR) at the receiver of UE i on channel c is given as:

$$\gamma_i(c) = \frac{P_i g_i(d)}{\sum_{j \in \mathscr{D}(c)\setminus i} P_j g_{j,i}(d') + N}, \tag{5.2}$$

where g_i is the channel power gain between UE i and its receiver, $g_{j,i}$ is the channel power gain between UEs i and j. The data rate $\nu_i(c)$ of UE i on the channel c of bandwidth W_c is obtained using the classical Shannon capacity formula, $\nu_i(c) = W_c \log_2(1+\gamma_i(c))$.

Note that the channel power gains g_i, $g_{i,j}$ are subject to random variations. These variations arise due to randomly varying channel gain, feedback errors, feedback delay errors, and quantization errors [59]. Therefore, all the quantities defined are in fact random variables. We denote $\hat{\nu}$ and ν as the estimated data rate and the expected data rate, respectively.

5.2.2 Problem Formulation

Our objective is to maximize the expected sum data rate of the D2D network by optimally assigning channels to UEs. Let $\bar{c} = (c_i, c_{-i})$ denotes a channel assignment vector where UE i is assigned the channel $c_i \in \mathscr{F}$ and UEs other than UE i are assigned the channel vector $c_{-i} \in \mathscr{F}^{|\mathscr{D}|-1}$. The estimated data rate of a UE depends on vector $\bar{c}$ and is denoted as $\hat{v}_j(\bar{c})$. The objective function is

$$\hat{\phi}(\bar{c}) = \sum_{j \in \mathscr{D}} \hat{v}_j(\bar{c}). \tag{5.3}$$

Let $\phi(\bar{c}) = \mathbb{E}\left[\hat{\phi}(\bar{c})\right]$ be expected value where the expectation is over all the randomness. Formally, CAP is stated as:

$$\bar{c}^* \in \underset{\bar{c} \in \mathscr{F}^{|\mathscr{D}|}}{arg\,max} \quad \phi(\bar{c}). \tag{5.4}$$

We seek to maximize the average sum data rate by using only estimates of data rates. This kind of problems where the expected value of a function is maximized is a class of SOPs [93]. In the next sections, we first model this SOP as a noisy-potential game and then solve it by adapting the developed BLLA in noisy-potential games.

5.3 Noisy Potential Game Framework

In real scenarios, UEs can have only their estimated throughput that is corrupted by noise. Therefore, we model the channel assignment problem (5.4) with noisy utilities.

Definition 14 *[CAP game] A CAP game is defined by the tuple* $\hat{\mathscr{G}} := \left\{\mathscr{D}, \{X_i\}_{i \in \mathscr{D}}, \{\hat{U}_i\}_{i \in \mathscr{D}}\right\}$, *where* $\mathscr{D}$ *is a set of UEs that are players of the game,* $\{X_i\}_{i \in \mathscr{D}}$ *are action sets consisting of orthogonal channels,* $\hat{U}_i : X \to \mathscr{R}$ *are random utility functions with finite expectation, and* $X = X_1 \times X_2 \times \ldots X_{|\mathscr{D}|}$.

An action profile $a := (a_i, a_{-i})$ where $a_i \in X_i$ is the action of player of i and $a_{-i} \in X_{-i}$ is the action set of all the players except player i. Note that the action vector $a \in X$ is the channel assignment vector $\bar{c}$ and $X = \mathscr{F}^{|\mathscr{D}|}$.

In general, the outcome of a game is some equilibrium like Nash Equilibrium (NE). However, our goal is to find the global maximizer of the CAP objective function rather than finding an NE. As discussed earlier, potential games are well suited for this purpose because the players can distributively optimize the potential function. Moreover, for potential games, an NE always exists and there exist algorithms that are guaranteed to converge to an NE or to the global maximizers of the potential function when utilities are deterministic [3]. We

directly cannot use the potential game model for our CAP game due to noisy utilities. To use the properties of potential games for CAP game we introduce noisy potential games. In the rest of this section, we illustrate the modelling of CAP game as a noisy potential game.

The utility functions of CAP game has to be designed carefully to model it as a noisy potential game. We consider the following utility function which represents the marginal contribution of the player to the global utility that is sum data rate.

$$\hat{U}_i(a_i, a_{-i}) = \sum_{j \in \mathscr{D}(a_i)} \hat{v}_j(a_i, a_{-i}) - \sum_{j \in \mathscr{D}(a_i) \setminus i} \hat{v}_j(a_{-i}), \tag{5.5}$$

where $\hat{v}_j$ is the measured data rate of player j and $\mathscr{D}(a_i) \subset \mathscr{D}$ is a set of players whose action is a_i i.e., $\mathscr{D}(a_i) = \left\{ j \in \mathscr{D} : a_j = a_i \right\}$.

Proposition 10 *A CAP game* $\mathscr{G} := \left\{ \mathscr{D}, \{X_i\}_{i \in \mathscr{D}}, \left\{\hat{U}_i\right\}_{i \in \mathscr{D}} \right\}$ *with utility functions (5.5) is a noisy potential game with potential function* $\phi(a)$.

Proof: Note that the utility function (5.5) is a sums of random data rates. Therefore, the expected utility can be expressed as the sum of expected data rates. The expected utility of player i represents the marginal contribution of the player to the expected global utility $\phi(a)$. It is well known that a game with marginal contribution utility functions leads to an exact potential game [20]. ■

As discussed in Chapter 2 the noisy utility $\hat{U}$ may have a large variance that leads a noisy potential game to have a large deviation from the exact potential game. To reduce the variance of the utility we use a sample mean of utility function as below

$$\hat{U}_i^N = \frac{1}{N} \sum_{k=1}^{N} \hat{U}_i. \tag{5.6}$$

Corollary 5 *A CAP game* $\hat{\mathscr{G}}^N := \left\{ \mathscr{D}, \{X_i\}_{i \in \mathscr{D}}, \left\{\hat{U}_i^N\right\}_{i \in \mathscr{D}} \right\}$ *is a noisy potential game with potential function* $\phi(a)$.

In the rest of the chapter, we consider the noisy potential CAP game $\hat{\mathscr{G}}^N$.

5.4 Distributed Learning Algorithm for CAP

In this section, we first describe the proposed binary log-linear algorithm (BLLA) in noisy potential games for distributively solving CAP.

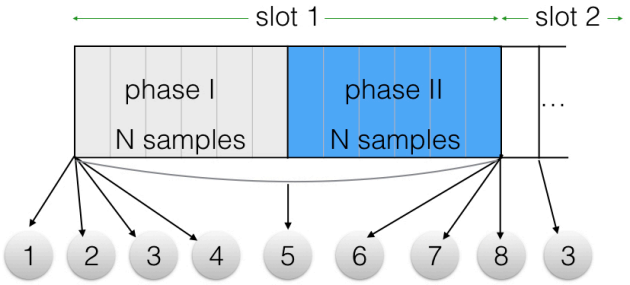


Figure 5.2: Time slots, phases, and steps of BLLA are shown in this figure. Steps are represented using circles. Step 5 is estimation duration of the UEs on two channels $a_i(t-1)$ and $\hat{a}_i$.

Algorithm 7 Binary Log-linear Learning Algorithm

1: **Initialisation:** Start with arbitrary channel assignment vector a.
2: **While** $t \geq 1$ **do**
3: Set parameter $\tau(t)$.
4: BS randomly selects a UE i and a trial channel $\hat{a}_i \in X_i$ with uniform probability. BS informs UE i and all the UEs with channels $a_i(t-1)$ and $\hat{a}_i$ to estimate their sample mean data rates.
5: UE i transmits on channels $a_i(t-1)$ and $\hat{a}_i$ during Phase I and Phase II, respectively.
6: At the end of Phase II, all UEs with channels $a_i(t-1)$ and $\hat{a}_i$ feedback to BS their two estimates of their sample mean data rates corresponding to Phases I and II.
7: BS calculates $\hat{U}_i^N(a(t-1))$, $\hat{U}_i^N(\hat{a}_i, a_{-i}(t-1))$, and selects a channel $\hat{a}_i$ with probability

$$\left(1+e^{\Delta_i^N/\tau}\right)^{-1}, \tag{5.7}$$

where $\Delta_i^N = \hat{U}_i^N(a(t-1)) - \hat{U}_i^N(\hat{a}_i, a_{-i}(t-1))$.
8: BS informs UE i to transmit on the selected channel. All the other UEs transmit on previous channels, i.e., $a_{-i}(t) = a_{-i}(t-1)$.

Table 5.1: Simulation parameters.

Parameter	Variable	Value
Number of orthogonal channels	$\mathscr{F}$	5
Channel bandwidth	W_c	180 KHz
Carrier frequency	f_c	2.6 GHz
Number of UEs	$\lvert\mathscr{D}\rvert$	20
Total transmit power of BS	P_{BS}	46 dBm
Transmit power of UE	P_{UE}	25 dBm
Minimum SINR	$\gamma_{\min}$	-10 dB
Maximum SINR	$\gamma_{\max}$	23 dB
Additive noise power per channel	N	$-174+10\log(W_i)$ dBm
Pathloss exponent	η	3.5
Shadowing variance	σ_{sh}	6

The details of BLLA for CAP are described in Algorithm 7 and shown in Fig. 5.2. Each time slot is divided into two phases of size N samples each. At the beginning of each time slot t, BS randomly selects a UE i and a trial channel $\hat{a}_i \in X_i$ with uniform probability. Also, BS informs all the UEs with channels $a_i(t-1)$ and $\hat{a}_i$ to estimate their data rates and feedback to BS at the end of two phases. UE i transmits on channel $a_i(t-1)$ and $\hat{a}_i$ during Phase I and Phase II, respectively. At the end of Phase II, all UEs on channels $a_i(t-1)$ and $\hat{a}_i$ feedback to BS their two estimates of their sample mean data rates corresponding to Phases I and II. BS calculates the utility of UE i according to (5.6) and selects a channel from the set $\{a_i(t-1),\hat{a}_i\}$ according to (5.7), where $\tau(t)$ is a temperature parameter that governs the convergence properties of BLLA. Then, BS informs UE i with the selected channel. This feedback requires only one bit. BLLA is distributed in nature because only a few UEs have to feedback to BS.

The convergence result of BLLA is given in Theorem 2. We use the sufficient number of samples given in Theorem 2 in simulations in the next section.

5.5 Simulations

In this Section, we present simulation results considering standard wireless system parameters shown in Table 5.1. We consider that a BS is located at the center of a region of radius 200 m. Among 20 UEs there are 5 UECs. The UECs have dedicated channels and no two UECs are on the same channel. These UECs serve as passive players of the game because they do not change their channel. The receivers of UED transmitters are located around

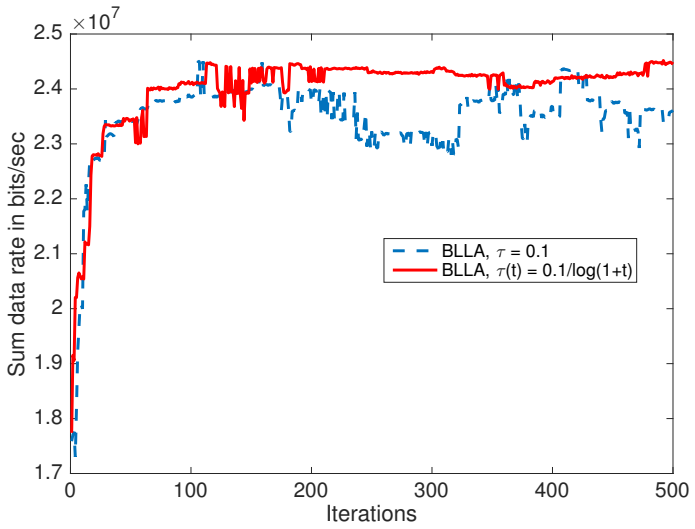


Figure 5.3: [Convergence of BLLA in downlink for fixed temperature and decreasing temperature]: It can be seen that the decreasing temperature results in smoother convergence compared to fixed.

them uniformly random over a region of radius 20 m. The UEDs learn their channel with the help of BS and hence are the active players of the game.

The variations of Rayleigh fading over time are considered as the noise component for all the simulations. These variations appear in the data rates of UEs. The data rates are bounded because the SINR is bounded between $\gamma_{\min}$ and $\gamma_{\max}$. We assume only bounded noise in the simulations where the noise interval $\ell = 1$ due to normalized utilities. Besides, the additive white Gaussian noise with power N is considered.

5.5.1 Downlink Sum Data Rate

In this subsection, we show the simulations for the downlink of the system. In Fig. 5.3, convergence to the maximum sum data rate of BLLA is shown with fixed temperature $\tau = 0.1$ and decreasing temperature $\tau(t) = 0.1/\log(1+t)$[1]. The number of samples are calculated according to (2.20) of Theorem 2 for $\tau = 0.1$ and $\xi = 10^{-5}$. BLLA reaches the maximum sum data rate with both fixed and decreasing temperatures. However, it has more variations for fixed temperature. For decreasing temperature, the probability of staying at the maximum is higher, which is evident from Fig. 5.3.

[1]Note that $\tau(t) = 0.1/\log(1+t)$ works well even though it is smaller than that given by Theorem 3. The reason being that the height of the highest local maximum is smaller than the global maximum [26]. We consider that height to be 10% of the global maximum, which is reasonable.

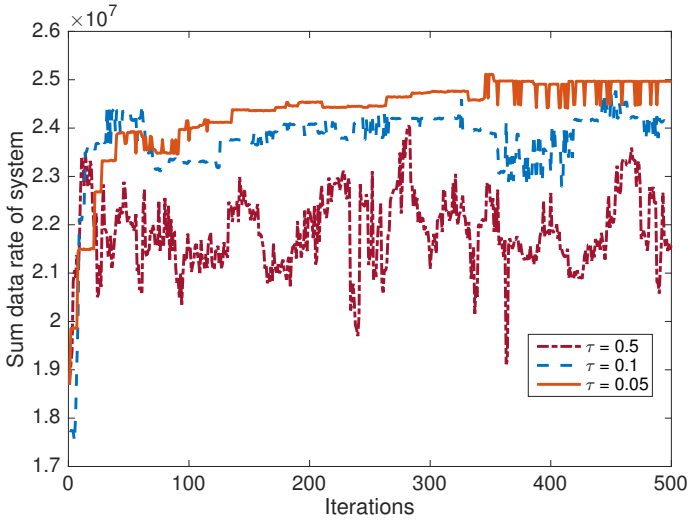


Figure 5.4: [Convergence of BLLA for different fixed temperature]: For $\tau = 0.5$, high fluctuations are observed compared to that for $\tau = 0.1$. BLLA with $\tau = 0.05$ gives more stable performance.

To study the effect of temperature we show the performance of BLLA for different temperatures in Fig. 5.4. As before, the samples are calculated corresponding to different temperatures. For higher temperature $\tau = 0.5$, BLLA exhibits huge variations. The probability of being at a local maximum decreases with increasing temperature. As temperature decreases, the variations also decrease. Also, the probability of being at a local maximum increases with decreasing temperature. Therefore, the temperature should be chosen carefully to obtain the desired performance. BLLA with smaller $\tau = 0.05$ gives the desired performance.

We study the effect of the number of samples on the performance of BLLA in Fig. 5.5. If players take decisions after every single sample, i.e., if the estimation errors are ignored, BLLA exhibits large variations due to noise. As the number of samples increases the performance of BLLA improves. If the number of samples is taken according to (2.20) of Theorem 2 then BLLA provides high and stable sum data rate.

We now study the effect of the number of channels in Fig. 5.6. This plot shows the average sum data rate obtained from BLLA at the end of 500 iterations averaged over 1000 realizations. We see that the sum data rate increases as the number of channels increases. This is intuitive because the optimal channel assignment has lower interference per channel. As evident from the figure, BLLA correctly tracks this phenomenon.

We also study the performance of BLLA by varying the number of UEs in Fig. 5.7 for 10 orthogonal channels and 10 UECs. As before, the sum data rate is obtained from

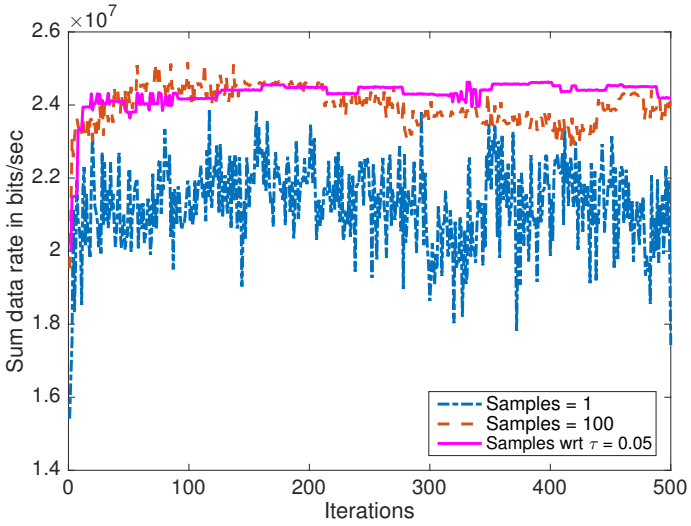


Figure 5.5: [Effect of number of samples on convergence of BLLA]: For $\tau = 0.05$, the number of samples N taken according to Theorem 2 gives smoother convergence. Otherwise, high fluctuations are observed.

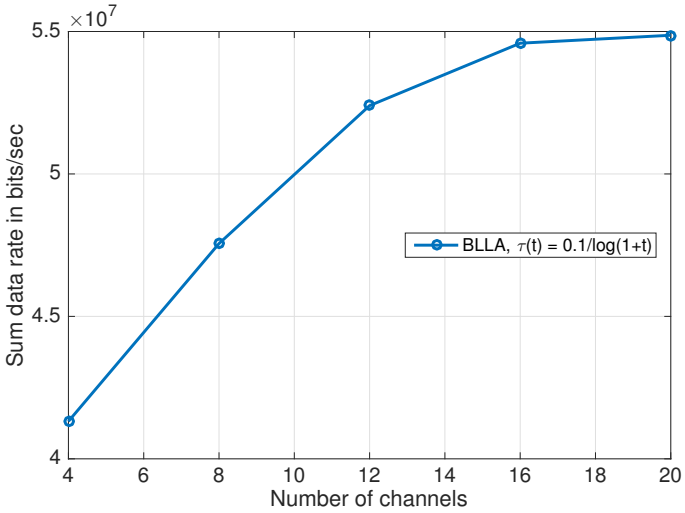


Figure 5.6: [Effect of a number of channels on sum data rate]: The number of UEs is fixed to 20. We see that the sum data rate increases with the number of channels since BLLA assign channels optimally leading to lower interference per channel.

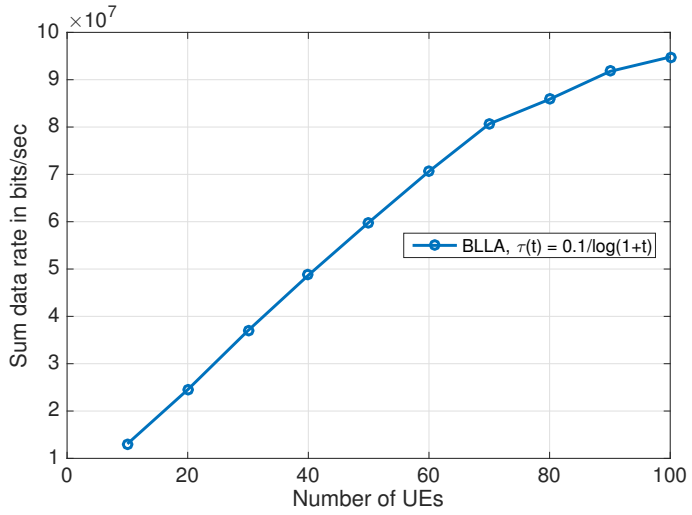


Figure 5.7: [Effect of number of UEs on sum data rate]: The number of channels is fixed to 10. We see that the sum data rate increases linearly until 60 UEs as BLLA manages to assign channels optimally and maintain low interference. For more number of UEs than 60 the interference per channel significantly increases that effects the sum data rate.

BLLA at the end of 500 iterations averaged over 1000 realizations. As the number of UEs increases (up to approximately 60 in the figure), the sum data rate first linearly increases because of the increasing traffic. A linear growth is observed as long as interference is controlled. Sixty is much larger than the number of available channels, which means that BLLA manages to assign frequencies in such a way that UEs do not interfere too much. After 60 UEs, the increase is reduced because interference significantly affects the sum data rate. BLLA exactly tracks this behaviour as evident from Fig. 5.7.

We now compare in Fig. 5.8 the performance of BLLA and better response (BR) algorithm. In better response algorithm, a player accepts the trial action with probability one if its utility is better than the current action unlike in BLLA. Best response algorithm, which is same as the better response with two actions, was applied to CAP with the objective of minimizing the total interference in Wireless Sensor Networks (WSN) in [94]. The parameter of BLLA is $\tau(t) = 0.1/\log(1+t)$. Each curve in the figure is obtained by averaging over 1000 realizations of the algorithms. BR performs the worst when noise is not taken into account, which corresponds to one estimation sample case. When the number of samples is increased to 200, BR improves. However, BLLA is better than BR. For 2000 samples, BR performance is the same as BLLA. It shows that the number of samples for BR has to be tuned carefully to obtain the desired performance. On the contrary, BLLA performs better with the fixed number of samples without any tuning. Note that

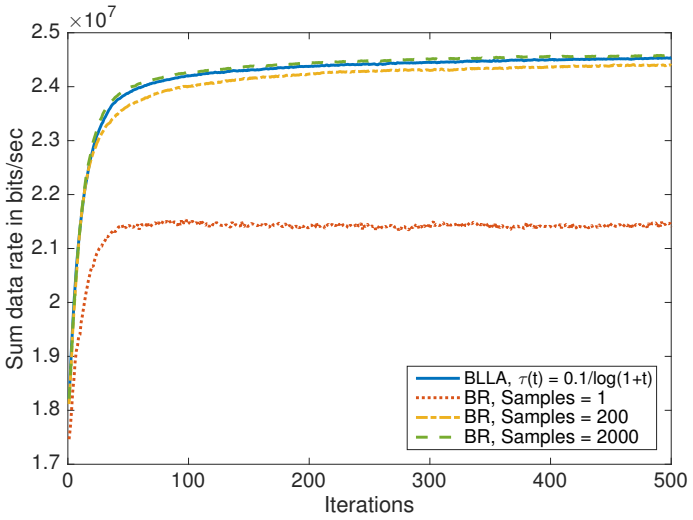


Figure 5.8: [Comparison of BLLA and better response algorithms]: BR with a single sample gives the worst performance. The performance of BLLA is better than BR for the number of samples 1 and 200. Only when the BR uses a very high number of samples of 2000 then its performance is same as that of BLLA.

theoretically, BR needs an infinite number of samples. There is no theoretical guarantee for its convergence for a finite number of samples. On the contrary, as we have proved BLLA has a theoretical guarantee of convergence with a finite number of samples.

5.5.2 Uplink Sum Data Rate

We also show that BLLA can be applied for channel assignment in the uplink of the network. The uplink sum data rate obtained from BLLA for fixed and decreasing temperatures are shown in Fig. 5.9. With fixed temperature, BLLA reaches the global maximum but leaves it quickly. However, with decreasing temperature BLLA stays at the global maximum for most of the time. This shows the applicability of BLLA for both uplink and downlink.

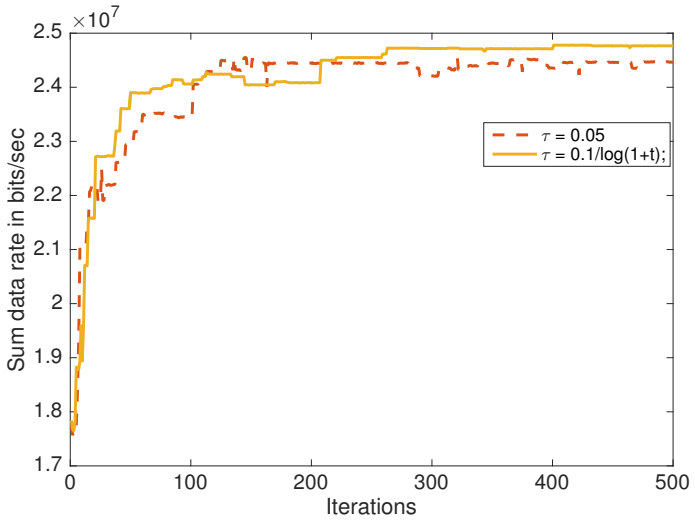


Figure 5.9: [Convergence of BLLA in uplink]: As in downlink we see that with $\tau(t) = 0.1/\log(1+t)$ the convergence of BLLA is smoother than that with fixed $\tau = 0.05$.

5.6 Conclusions

A distributed learning algorithm BLLA for CAP in D2D wireless networks is presented in this chapter. To capture the throughput estimation noise a noisy potential game framework is used. BLLA achieves the optimal channel assignments that maximize the sum data rate of the network. A sufficient number of estimation samples that guarantees the convergence in both cases of bounded noise and unbounded noise given by Theorem 2 are validated using simulations. The sum data rate increases with the number of channels and number of UEs. BLLA performs better than the better response algorithm in the presence of noise.

Chapter 6

Distributed Greedy Algorithm for Submodular Maximization

In this chapter, we consider a different setting other than potential games. We consider the distributed maximization of a submodular function, which is a set function that has the property that the difference of the function decreases with the increasing size of a given set when an element is added to it. The maximization of submodular functions is a well-studied topic due to its application in many common engineering problems. Because this problem has been shown to be NP-Hard for certain subclasses of functions, much work has been done to develop efficient algorithms to approximate an optimal solution. Among these is a simple greedy algorithm, which has been shown to guarantee a solution within $\frac{1}{2}$ the optimal. However, when this algorithm is implemented in a distributed way, it requires all agents to share information with one another - a costly constraint for some applications. In this chapter, we explore how the degradation of information sharing among the agents affects the performance of the distributed greedy algorithm. For any underlying communication graph structure, we show results for how well the distributed greedy algorithm can perform. In addition, for applications where the number of agents and number of communication links is fixed, we identify near-optimal graph structures with the highest performance guarantees. This result can inform system designers as to the most impactful places to insert communication links.

6.1 Introduction

The optimization of submodular functions is a well-studied topic due to its application in many common engineering problems. Examples include information gathering [95], maximizing influence (viral marketing) in social networks [96], image segmentation in image processing [97], multiple object detection in computer vision [98], document summarization [99], path planning of multiple robots [100], sensor placement [101, 102], and resource allocation in multi-agent systems [102]. For each case, the problem can be formulated as the optimization of a submodular function.

While polynomial algorithms exist to solve submodular minimization, [103–105], maximization has been shown to be NP-Hard for certain subclasses of submodular functions [106]. Thus a tremendous effort has been placed on developing fast algorithms that approximate the solution to the submodular maximization problem [107–113]. The resounding message from this extensive research is that very simple algorithms can provide strong guarantees on the quality of the approximation. Specifically, this work focuses on submodular maximization with a matroid constraint. Using a greedy randomized approach, [111] shows that the optimal solution can be approximated within a factor of $1 - 1/e \approx 0.63$.

Another proposed technique is to use a simple greedy algorithm, which has been shown to guarantee an approximate solution within $1/2$ of the optimal [107]. One reason this algorithm remains relevant is that it lends itself to a distributed approach to maximization. In this approach, the distributed greedy algorithm, a set of agents sequentially maximizes a function without the need of centralized information. The result is simply the compilation of all the agents' decisions. As each agent maximizes its function, however, it must be able to evaluate all its potential choices and have access to the decisions of all previous agents. In many multi-agent systems, this amount of information and sharing is costly.

Research has therefore begun to explore how limited information and sharing can impact the performance of the distributed greedy algorithm. For example, [114] explores the performance when an agent can only evaluate a local subset of its choices. The work in [115] studies the performance when an agent can only observe a local subset of its predecessors. In each case, results show that localizing information can substantially degrade performance.

This work more closely relates to the work done in [115] in evaluating information sharing constraints. Their work models communication structures as graphs and demonstrates how performance can degrade for certain graphs. For instance, if agents are partitioned into groups that only transmit among themselves, the lower bound on performance degrades by

a factor proportional to the number of groups. However, two questions remain open:

1. For any communication structure, what are the performance guarantees?
2. What communication structures make the highest performance guarantees?

The contribution of this work is to provide insights into answering these two questions. More specifically

1. Theorem 11 gives lower and upper bounds on worst-case performance for a given communication structure. The bounds give intuition on what properties of the underlying graph need to be improved in order for performance to improve.
2. Theorem 12 shows the best performance that a system designer can achieve with a fixed number of agents and communication links. The results show that when information is costly, the best communication structures spread out the communication links among the agents, rather than clustering them among a small group.

The remainder of this chapter is dedicated to proving these two theorems.

6.2 Model

This chapter focuses on the design of distributed algorithms for attaining desirable solutions for submodular maximization. To that end, let S be a set of elements and $f : 2^S \to \mathbb{R}_{\geq 0}$ have the following properties:

- *Submodular*: For $A \subseteq B \subset S$ and $x \in S \setminus B$, the following holds:

$$f(A \cup \{x\}) - f(A) \geq f(B \cup \{x\}) - f(B). \tag{6.1}$$

- *Monotonic*: For $A \subseteq B$, $f(A) \leq f(B)$.
- *Normalized*: $f(\emptyset) = 0$.

We say f is *modular* if (6.1) is an equality.

In many applications maximizing such f can be formulated as the compilation of decisions made by agents, each maximizing a local utility. Suppose we have agents labeled $1, ..., n$ and a corresponding family of sets $X_1, ..., X_n$, where $X_i \subseteq 2^S$. Here we define $X = X_1 \times \cdots \times X_n$. For notational purposes, if A is a set of agents, then let $x_A = \bigcup_A x_i$, where

$x_i \in X_i$ and $i \in A$. We also use the notation that $x_{a:b} = \bigcup_{i=a}^{b} x_i$ for $b \geq a$. Finally, in the following, we define $x_1^*, ..., x_n^*$ as the optimal set of decisions for each agent. In this context, the submodular maximization problem can be formulated as

$$f(x_{1:n}^*) = \max_{x_1 \in X_1, ..., x_n \in X_n} f(x_{1:n}). \tag{6.2}$$

This type of constraint where we choose from a family of subsets X_i is equivalent to a partition matroid constraint [115]. Each X_i can be considered the action space of agent i.

As mentioned in the previous section, the maximization problem in (6.2) can be approximated using a distributed greedy algorithm. Since the algorithm requires agents to make decisions sequentially, without loss of generality we impose an ordering on the agents according to their labels, i.e., agent 1 chooses first, agent 2 chooses second, etc. Agent i makes its choice $x_i \in X_i$ based on the following rule:

$$x_i \in arg\,max_{\tilde{x}_i \in X_i} f(\tilde{x}_i \cup x_{1:i-1}), \tag{6.3}$$

where $x_{1:0}$ is defined to be $\emptyset$. Note that there could be several x_i to choose from in the $arg\,max$, so define $\mathscr{X} \subseteq X$ to be the set of all possible choices if agents choose according to (6.3). It is well-known in the literature that for any submodular f, any set of action spaces X, and any order of agents, the quality of the resulting solution $x = x_{1:n}$ derived from the distributed greedy algorithm compared to the optimal solution $x^* = x_{1:n}^*$ satisfies

$$\gamma(f, X) := \frac{\inf_{x \in \mathscr{X}} f(x)}{f(x^*)} \geq \frac{1}{2}, \tag{6.4}$$

which means the solution is within 50% of the optimal [107]. Here we say that $\gamma(f, X)$ is the efficiency of solution x. For special classes of submodular functions, and additional constraints placed on $X_1, ..., X_n$, [107] also shows that $\gamma(f, X) \geq 1 - 1/e$. Note that this greedy algorithm is different compared to the best response algorithm, in which the current agent has to know the action of all the agents.

Notice that in (6.3), agent i must have access to the decisions of all previous agents. However, this may not hold in some real-world applications. Therefore a more generalized version of the distributed greedy algorithm is where agent i makes its choice using the following rule:

$$x_i \in arg\,max_{\tilde{x}_i \in X_i} f(\tilde{x}_i \cup x_{\mathscr{N}_i}), \tag{6.5}$$

where $\mathscr{N}_i \subseteq \{1,...,i-1\}$. The sets $\mathscr{N}_i$, $i = 1,...,n$ are the sets that represent the communication structure - in other words, $\mathscr{N}_i$ is the set of agents whose choices are observed by agent i when making its decision. It is helpful to visualize this communication structure as a directed graph $G = (V,E)$, where V is a set of vertices and $E \subseteq V \times V$ is a set of edges between vertices. In this scenario each vertex is an agent and each edge (j,i) implies $j \in \mathscr{N}_i$, i.e., $\mathscr{N}_i$ is the set of in-neighbors for vertex i. Since there is an imposed ordering on the vertices, and the agents choose sequentially, the set

$$\mathscr{G} = \{G = (V,E) : (i,j) \in E \implies i < j\} \tag{6.6}$$

is the set of admissible graphs that correspond to a communication structure. In this context, we define $\mathscr{X} \subseteq X$ to be the set of all possible choices if agents choose according to (6.5), and again measure the efficiency of solution x according to the decision rule (6.5) with

$$\gamma(f,X,G) := \frac{\inf_{x \in \mathscr{X}} f(x)}{f(x^*)}, \tag{6.7}$$

where $G \in \mathscr{G}$ is defined by the sets $\mathscr{N}_i$.

The goal of this work will be to identify the efficiency guarantees associated with this more generalized version of the distributed greedy algorithm for any monotone normalized submodular function. To that end, let $\mathscr{F}$ be the set of such functions, where each $f \in \mathscr{F}$ is defined on some element set S_f. We now define

$$\gamma(G) = \inf_{f \in \mathscr{F}} \inf_{X : X_i \subseteq 2^{S_f}} \gamma(f,X,G) \tag{6.8}$$

In words, $\gamma(G)$ is the worst-case efficiency for any f and family of sets $X_1,...,X_n$ as defined above, given that the communication structure among the agents is represented by G.

6.2.1 An Illustrative Example: Weighted Set Cover

We now describe the weighted set cover problem using our model, in order to illustrate how the generalized distributed greedy algorithm works using a communication structure represented by a graph G. The following sections leverage instances of this problem to show worst-case scenarios and to prove tight lower bounds, so it serves as both a simple example and part of our proof technique. For a base set of elements S:

- let $s_1,...,s_T$ be a partition on S, each with an associated positive value $v_1,...,v_T$,

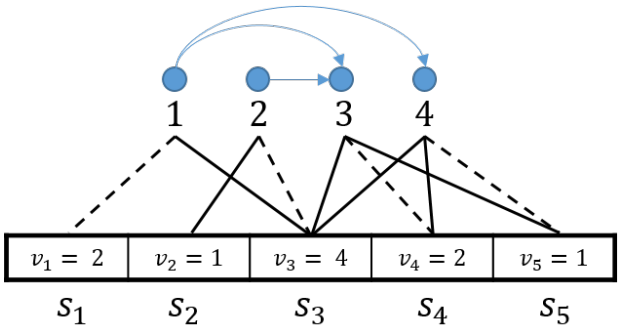


Figure 6.1: The setup of a weighted set cover problem. The set S is partitioned into subsets $s_1, ..., s_5$, each with a corresponding value $v_1, ..., v_5$. The available choices to each agent are represented by the black lines, where the dotted lines are an optimal set of choices. The goal for the agents is to maximize the value of the union their choices.

Algorithm	x_1	x_2,	x_3	x_4	$f(x_{1:4})$
Optimal	s_2	s_3	s_4	s_5	9
Distributed Greedy	s_3	s_1	s_4	s_5	8
Generalized Distributed Greedy	s_3	s_3	s_4	s_4	6

Figure 6.2: This figure shows the agents' decisions in an optimal case, the case where the distributred greedy algorithm is used (agents choose according to (6.3)) and the case where the generalized distributed algorithm is used (agents choose according to (6.5), constrained to the graph shown in Fig. 6.1).

- let $X_i \subseteq \{s_1, ..., s_T\}$, and
- let $f(x_{1:n}) = \sum_{j \in J} v_j$, where $j \in J$ if s_j was chosen by some agent.

In this case, f is submodular (more specifically, it is modular), monotone, and normalized, thus it meets the requirements for the model. Essentially, the agents are collectively trying to "cover" as much of S as they can, given each agent i's restriction to choose from X_i. An instance of the weighted cover set problem is shown in Talbe 6.2.

6.3 Worst-Case Efficiency Bounds

In this section we present lower and upper bounds for the worst-case efficiency $\gamma(G)$ for any $G \in \mathscr{G}$ based on its structure. We begin with some preliminaries from graph theory, and then prove the bounds.

6.3.1 Preliminaries

In order to show bounds on $\gamma(G)$, we leverage terminology and work done in graph theory. For all definitions in this section, we assume that $G = (V, E)$ is any general directed graph. We begin with cliques:

- A *clique* is a set of nodes $C \subseteq V$ such that for every $i, j \in C$, either $(i, j) \in E$ or $(j, i) \in E$.
- The *clique number* $\omega(G)$ is the size of the largest clique in G.
- A *clique cover* is a partition on V such that the nodes in each partition form a clique.
- The *clique cover number* $k(G)$ is the minimum number of sets within any clique cover of G.

As an example, consider the graph in Fig. 6.3. Here there are 4 cliques of size 1 (one for each node), 5 cliques of size 2 (one representing each edge), and 2 cliques of size 3 (the sets $\{1,2,3\}$ and $\{1,2,4\}$). Thus the clique number is 3. A minimum clique cover is $\{1,3\}, \{2,4\}$, so the clique cover number is 2.

Another important notion in graph theory is that of independence:

- An *independent set* $I \subseteq V$ is a set of vertices such that $v_1, v_2 \in I$ implies $(v_1, v_2), (v_2, v_1) \notin E$.

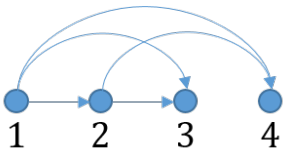


Figure 6.3: An example graph used to illustrate some basic graph theory concepts. In this case $\omega(G) = 3$, $k(G) = 2$, and $\alpha(G) = 2$.

- A *maximum independent set* $I_{\max}$ is an independent set of G such that no other independent set has more vertices.
- The *independence number* of G is $\alpha(G) := |I_{\max}|$.

As an example, consider again the graph in Fig. 6.3. The maximum independent set is $\{3,4\}$, thus the independence number is 2.

6.3.2 Related Work

As mentioned in the previous section, this work relates to that done in [115]. We give a brief description of their results, as it pertains to this section. The first is that for a graph $G \in \mathscr{G}$:

$$\gamma(G) \geq \frac{1}{n - \omega(G) + 2}, \tag{6.9}$$

and a second is that if G is a family of disconnected cliques, then

$$\gamma(G) \geq \frac{1}{2k(G)}. \tag{6.10}$$

The lower bound on $\gamma(G)$ that we show in the next section is greater than or equal to these bounds in all cases.

6.3.3 Lower and Upper Bounds on Efficiency $\gamma(G)$

Now we present lower and upper bounds on $\gamma(G)$ in the terms of the clique cover number $k(G)$ and independence number $\alpha(G)$, respectively. In order to assist, let

$$\Delta(x_i|x_P) = f(x_i, x_P) - f(x_P) \tag{6.11}$$

be the marginal contribution of agent i given the choices of the agents in set P. We will also make use of the following Lemma:

Lemma 13 *Let $A \subseteq S$ and $B \subseteq V$. Then*

$$f(A,x_B) = f(A) + \sum_{i \in B} \Delta(x_i|A, x_{j \in B: j<i}). \tag{6.12}$$

Proof: Let j be the smallest element of B. From (6.11)*, it follows that $f(A,x_B) = f(A,x_{B\setminus\{j\}},x_j) = f(A,x_{B\setminus\{j\}}) + \Delta(x_j|A,x_{B\setminus\{j\}})$. Let l be the next smallest element of B, then it follows that*

$$\begin{aligned} f(A,x_B) =& f(A,x_{B\setminus\{j,l\}}) + \Delta(x_j|A,x_{B\setminus\{j\}}) \\ &+ \Delta(x_l|A,x_{B\setminus\{j,l\}}). \end{aligned} \tag{6.13}$$

Continuing this line of reasoning sequentially for all $i \in B$, we arrive at (6.12) ■

We now proceed with the main theorem and its proof.

Theorem 11 *Let $G \in \mathscr{G}$. Then*

$$\frac{1}{\alpha(G)} \geq \gamma(G) \geq \frac{1}{k(G)+1}. \tag{6.14}$$

Proof: We first show the lower bound beginning with the following inequality:

$$f(x^*_{1:n}) \leq f(x^*_{1:n}, x_{1:n}), \tag{6.15}$$

$$= f(x_{1:n}) + \sum_{i=1}^{n} \Delta(x^*_i|x_{1:n}, x^*_{1:i-1}), \tag{6.16}$$

$$\leq f(x_{1:n}) + \sum_{i=1}^{n} \Delta(x^*_i|x_{\mathscr{N}_i}), \tag{6.17}$$

$$\leq f(x_{1:n}) + \sum_{i=1}^{n} \Delta(x_i|x_{\mathscr{N}_i}), \tag{6.18}$$

where (6.15) *holds by monotonocity,* (6.16) *holds by Lemma 13,* (6.17) *holds by submodularity since $\mathscr{N}_i \subseteq \{1,...,n\}$, and* (6.18) *holds because agents make decisions according to* (6.5)*.*

Now let $C_1,...,C_k$ be a minimal clique cover of G. In other words, $C_1,...,C_k$ is a partition on V where for every $j \in \{1,...,k\}$ every vertex in C_j is connected to every other vertex in C_j. Let P be the function that maps vertex i to its assigned partition, i.e., if $i \in C_j$, then

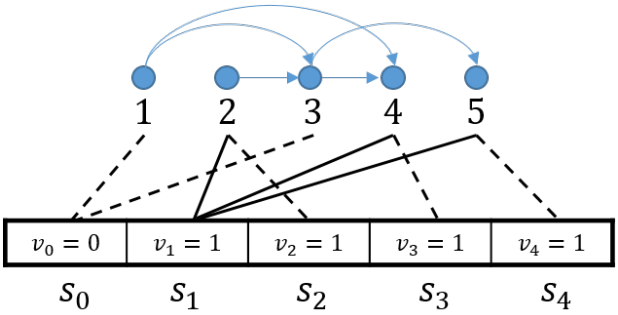


Figure 6.4: An example weighted set cover problem from the proof of Theorem 11. Here we use $I_{\max} = \{2,4,5\}$. The optimal choices are shown by dotted lines, thus $f(x^*_{1:5}) = 3$. The worst case using the generalized distributed greedy algorithm, is where 2, 4, and 5 all choose s_1, where $f(x_{1:5}) = 1$. Therefore, in this case $\gamma(f,X,G) = \frac{1}{\alpha}$.

$P(i) = j$. *We also define* $Q_i = \{m \in C_{P(i)} : m < i\}$. *Now we see that*

$$\sum_{i=1}^{n} \Delta(x_i|x_{\mathcal{N}_i}) \leq \sum_{i=1}^{n} \Delta(x_i|x_{Q_i}), \tag{6.19}$$

$$= \sum_{j=1}^{k(G)} \sum_{i \in C_j} \Delta(x_i|x_{Q_i}), \tag{6.20}$$

$$= \sum_{j=1}^{k(G)} f(x_{C_j}), \tag{6.21}$$

$$\leq \sum_{j=1}^{k(G)} f(x_{1:n}), \tag{6.22}$$

$$= k(G) f(x_{1:n}), \tag{6.23}$$

where (6.19) *is true by submodularity since* $Q_i \subseteq \mathcal{N}_i$, (6.20) *is true since it merely imposes a different order on the sum,* (6.21) *is true by Lemma 13,* (6.22) *is true by submodularity since* $C_j \subseteq \{1,\ldots,n\}$, *and* (6.23) *is the result of the sum. Substituting the above value into* (6.18), *we see that*

$$f(x^*_{1:n}) \leq (k(G)+1) f(x_{1:n}), \tag{6.24}$$

which holds for any f, S, *and* $X_1,\ldots,X_n$. *Therefore* $\gamma(G) \geq \frac{1}{k(G)+1}$.

Next we prove the upper bound. It is sufficient to show that for any G *we can choose* S, f *and* $X_1,\ldots,X_n$ *such that* $\gamma(f,X,G) = \frac{1}{\alpha(G)}$. *Then by definition* $\gamma(G)$ *cannot be greater than* $\gamma(f,X,G)$. *Consider a weighted set cover problem where* S *is partitioned by* $s_0, s_1, \ldots, s_{\alpha(G)+1}$, *where* $v_0 = 0$, *and where* $v_1 = \cdots = v_{\alpha(G)+1} = 1$. *Suppose that* $I_{\max}$ *is a*

maximum independent set in G, and that the action sets are assigned as follows

$$X_i = \begin{cases} \{s_1, P(i)\} & \text{if } i \in I_{\max} \\ \{s_0\} & \text{otherwise,} \end{cases}$$

where $P : I_{\max} \to \{s_2, ..., s_{\alpha(G)+1}\}$ *is injective. An instance of this scenario is shown in Fig. 6.4. Essentially, this choice of* X_i *has "zeroed out" any vertex that is not in* $I_{\max}$*, meaning that only vertices in* $I_{\max}$ *are adding value. However, note that by definition no vertex in* $I_{\max}$ *is connected to any other, so the corresponding agents make choices independently. The optimal set of choices are for each agent* $i \in I_{\max}$ *to choose* $P(i)$*, and thus* $f(x^*_{1:n}) = \alpha(G)$*. The worst set of choices would be for each agent* $i \in I_{\max}$ *to choose* s_1*, which they have equal incentive to do. In this case,* $f(x_{1:n}) = 1$*, thus* $\gamma(f, X, G) = \frac{1}{\alpha(G)}$*.* ■

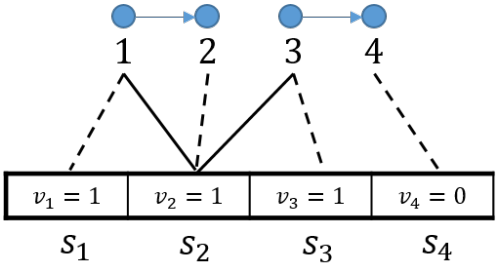
As stated, the lower bound presented in Theorem 11 is greater than or equal to those lower bounds shown in [115]. This is trivially true for (6.10), and it's true for (6.9) if $n - \omega(G) + 1 \geq k(G)$ for all $G \in \mathscr{G}$. We can see that this statement holds since $n - \omega(G)$ is all vertices outside a largest clique and $k(G)$ would not include any two from the same clique.

6.3.4 Examples

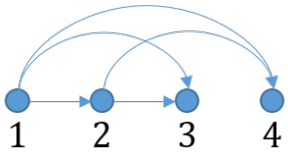
Theorem 11 shows lower and upper bounds on $\gamma(G)$, but we have not shown whether either of these bounds is tight. In fact, there exist some choices of f, S, and $X_1, ..., X_n$ where the lower bound is tight and other choices where the upper bound is tight. In this section, we provide an example of each.

Example 1 *Consider the weighted set coverage problem presented in Fig. 6.5a. For this graph G,* $\alpha(G) = k(G) = 2$*. As shown,* $\gamma(f, X, G) = \frac{1}{k(G)+1}$*. Since* $\gamma(G) \leq \gamma(f, X, G)$ *by definition, it follows that* $\gamma(G) = \frac{1}{k(G)+1}$*. Also, since* $\alpha(G) = 2$*, the upper bound in Theorem 11 is* $\frac{1}{2}$*, and is not tight in this case. As a note, the lower bounds on* $\gamma(G)$ *in* (6.9) *and* (6.10) *are both* $\frac{1}{4}$*.*

Example 2 *Consider the graph G in Figure 6.5b. Again* $k(G) = \alpha(G) = 2$*, and we claim the upper bound given in Theorem 11 is tight. This can be shown beginning with the*



(a) An example of a graph G where $\gamma(G) = \frac{1}{k(G)+1}$. Here $k(G) = 2$, and we can see that $f(x^*_{1:4}) = 3$. The worst-case results from the generalized distributed greedy algorithm occur when $x_1 = x_2 = x_3 = s_2$, and therefore $f(x_{1:4}) = 1$. Therefore $\gamma(f,X,G) = \frac{1}{k(G)+1} = \frac{1}{3}$, so the lower bound in Theorem 11 is tight for this graph.



(b) An example of a graph where $\gamma_{min} = \frac{1}{\alpha}$

Figure 6.5: Example graphs that meet the (a) lower bound and (b) upper bound given in Theorem 11.

following inequality:

$$
\begin{aligned}
f(x^*_{1:4}) \leq & f(x^*_{1:4}, x_{1:2}) && (6.25)\\
= & f(x_{1:2}) + \Delta(x^*_1|x_{1:2}) + \Delta(x^*_2|x^*_1, x_{1:2}) \\
& + \Delta(x^*_3|x^*_{1:2}, x_{1:2}) + \Delta(x^*_4|x^*_{1:3}, x_{1:2}) && (6.26)\\
\leq & f(x_{1:2}) + f(x^*_1) + \Delta(x^*_2|x_1) + \Delta(x^*_3|x_{1:2}) \\
& + \Delta(x^*_4|x_{1:2}) && (6.27)\\
\leq & f(x_{1:2}) + f(x_1) + \Delta(x_2|x_1) + \Delta(x_3|x_{1:2}) \\
& + \Delta(x_4|x_{1:2}) && (6.28)\\
= & f(x_{1:3}) + f(x_{1:2}, x_4) && (6.29)\\
\leq & 2f(x_{1:4}), && (6.30)
\end{aligned}
$$

where (6.25) *is true by monotonocity,* (6.26) *is true by Lemma 13,* (6.27) *is true by submodularity,* (6.28) *is true since agents choose according to* (6.5)*,* (6.29) *is true by Lemma 13, and* (6.30) *is true by submodularity. Therefore we conclude* $\gamma(G) = \frac{1}{2}$*, which is the upper bound given in Theorem 11. We also see that since* $k(G) = 2$*, the lower bound from Theorem 11 is* $\frac{1}{3}$ *and is therefore not tight.*

6.4 Near-Optimal Communication Graph

The bounds on $\gamma(G)$ given above dictate a near-optimal approach to communication structure design. Using these results, we describe how to build graph structures that approximate the highest $\gamma(G)$. In other words, given n vertices and m edges, we show where to put edges in order to maximize $\gamma(G)$. We begin by defining the following notation:

- Let $\mathscr{G}_{m,n} := \{G = (V,E) \in \mathscr{G} : |V| = n, |E| = m\}$.
- Let
$$G^*_{m,n} \in arg\,max_{G \in \mathscr{G}_{m,n}} \gamma(G). \tag{6.31}$$
Note that $G^*_{m,n}$ is not necessarily unique for a given m and n.
- For a graph $G = (V,E)$, its *complement* is $\bar{G} = (V,\bar{E})$, where $(i,j) \in E \iff (i,j) \notin \bar{E}$.

Theorem 12 *For any integers $m \geq 0$ and $n > 0$ such that $m \leq \frac{1}{2}n(n-1)$,*

$$\frac{1}{r^*} \geq \gamma(G^*_{m,n}) \geq \gamma(\hat{T}(n,r^*)) = \frac{1}{r^*+1} \tag{6.32}$$

where

$$r^* = \left\lceil \frac{n^2}{2m+n} \right\rceil \tag{6.33}$$

and $\hat{T}(n,r) \in \mathscr{G}$ for any positiver integer r is a graph constructed with the following algorithm:

1. *Partition the vertices into r different sets $C_1,...,C_r$ such that $|C_i|$ and $|C_j|$ differ by no more than 1 for all $i,j \in \{1,...,r\}$. In other words, all sets in the partition are as close to equal size as possible.*
2. *Create edges between all nodes within each set.*

Prior to the proof, we give some insight into the implication of this theorem. With some algebraic manipulation, (6.32) becomes

$$1 \geq \frac{\gamma(\hat{T}(n,r^*))}{\gamma(G^*_{m,n})} \geq \frac{r^*}{r^*+1}. \tag{6.34}$$

This format shows that the worst-case efficiency of $G^*_{m,n}$ can be approximated by that of $\hat{T}(n,r^*)$. The approximation gets closer as r^* increases, which according to (6.33)

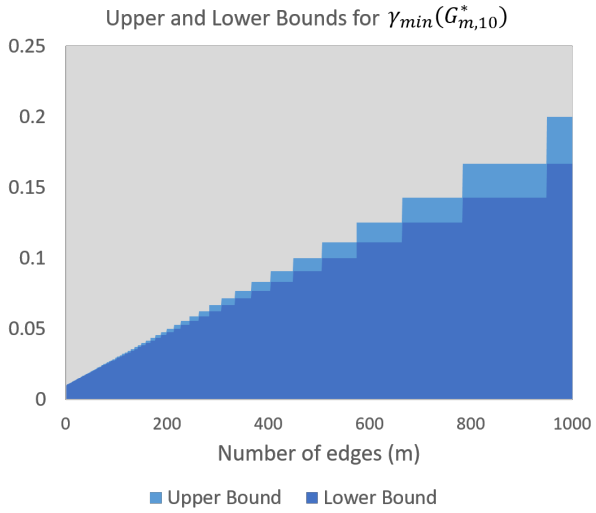
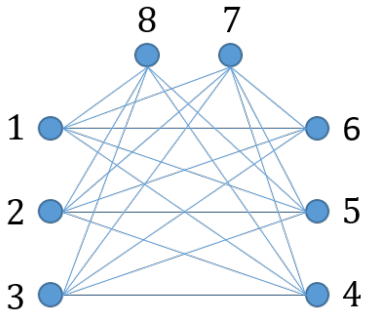
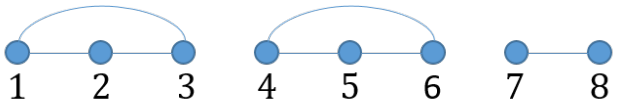


Figure 6.6: The lower and upper bounds shown in Theorem 12 for graphs with 100 vertices, ranging from 0 to 1000 edges.



(a) The Turán graph $T(8,3)$, where the clique number is 3. No other graph with 8 vertices can have more edges without also having a clique of size 4 or higher.



(b) The complement Turán graph $\hat{T}(8,3)$, where the independence number is 3. No other graph with 8 vertices can have less edges without also having an independent set of size 4 or higher.

Figure 6.7: A depiction of Turán graph in Fig. 6.7a and its complement in Fig. 6.7b is shown in this figure.

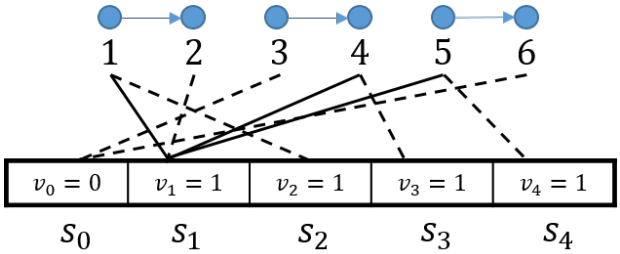


Figure 6.8: A graph $\hat{T}(6,3)$ with a weighted set cover problem described in the proof for Theorem 12, where we know $k(\hat{T}(6,3)) = 3$. In this instance, $y = 1$, $z = 2$ and the set $I = \{2,4,6\}$. By the dotted lines we see that the optimal choices yield $f(x^*_{1:6}) = 4$ and in the worst-case, the generalized distributed greedy algorithm yields $f(x_{1:6}) = 1$. Therefore $\gamma(f,X,\hat{T}(6,3)) = \frac{1}{4}$, and the lower bound on $\gamma(\hat{T}(6,3))$ given in Theorem 11 is tight.

corresponds to when m is on the order of n and not n^2. In other words, r^* is high for sparse graphs. This idea is represented in Fig. 6.6.

This result gives insight into how to construct a graph given n vertices and m edges: construct the graph $\hat{T}(n,r^*)$. As an example, if a system designer had 8 agents but could only place 7 edges, an optimal graph would be that in Fig. 6.7b, which is a complement graph of Turán shown in Fig. 6.7a. Implied in the following proof is that in the case where n is divisible by r^*, $G^*_{m,n} = \hat{T}(n,r^*)$. However, when this is not true, $\hat{T}(n,r^*)$ is an approximation of $G^*_{m,n}$, since $G^*_{m,n}$ has more edges. Theorem 11 does not give insight as to where to place these extra edges. Intuitively, this leads to the slack between the upper and lower bounds in Theorem 12, although technically it need not be the case that $\hat{T}(n,r^*)$ is an induced subgraph of $G^*_{m,n}$. We now proceed with the proof.

Proof: We first describe some properties of $\hat{T}(n,r)$ as is it constructed in the theorem statement. In graph theory a *Turán graph* $T(n,r)$ is a graph with n vertices created with the following algorithm:

1. Partition the vertices into r different sets $C_1,...,C_r$ such that $|C_i|$ and $|C_j|$ differ by no more than 1 for all $i,j \in \{1,...,r\}$.

2. Create edges between all nodes not within the same set.

The number of edges $\bar{m}$ in $T(n,r)$ satisfies the following inequality

$$\bar{m} \leq \left(1 - \frac{1}{r}\right) \cdot \frac{n^2}{2}. \tag{6.35}$$

A result known as Turán's theorem states that $T(n,r)$ is an n-vertex graph with the utmost

number of edges that has clique number r or smaller [116]. Alternatively stated,

$$T(n,r) \in arg\,max_{G=(V,E):\omega(G)\leq r}|E|. \tag{6.36}$$

Note that Step 1 above is the same as in the procedure for creating $\hat{T}(n,r)$, but Step 2 is different. Any edge in $T(n,r)$ will not be present in $\hat{T}(n,r)$, and vice versa, thus these graphs are complements of each other, i.e.,

$$\hat{T}(n,r) = \overline{T(n,r)}. \tag{6.37}$$

It is also well-known in graph theory that an independent set of vertices in a graph G forms a clique in $\bar{G}$. This implies that $\alpha(G) = \omega(\bar{G})$. Combining this fact with (6.37) and (6.36), we conclude that

$$\hat{T}(n,r) \in arg\,min_{G=(V,E):\alpha(G)\leq r}|E|. \tag{6.38}$$

In words $\hat{T}(n,r)$ is a graph with the least edges that has independence number r or fewer. An example of a Turán graph and its complement is found in Fig. 6.7.

We now show the rightmost equality in (6.32). By construction $\hat{T}(n,r)$ is a series of r disconnected cliques $C_1,...,C_r$, therefore $k(\hat{T}(n,r)) = r$. By Theorem 11 it follows that $\gamma(\hat{T}(n,r)) \geq \frac{1}{r+1}$. However, to show that this bound is tight, we again leverage the weighted set cover problem. Let y and z be the first and second vertices, respectively, in some clique. Let $s_0,...,s_{r+1}$ be the partition on S, where $v_0 = 0, v_1 = \cdots = v_{r+1} = 1$, and let I be a set of nodes, one from each clique C_j, that includes z. Then the X_i are assigned as follows:

$$X_i = \begin{cases} \{s_1\} & \text{if } i = z \\ \{s_1, P(i)\} & \text{if } i \in I \\ \{s_0\} & \text{otherwise,} \end{cases}$$

where $P : I \to \{s_2,...,s_{r+1}\}$ is injective. An instance of such a weighted set cover problem is shown in Fig. 6.8. Here the optimal sequence is for each $i \in I$ to choose $P(i)$, yielding $f(x^*_{1:n}) = r+1$. However, the worst-case generalized distributed greedy algorithm set of decisions is when all $i \in I$ choose s_1, so that $f(x_{1:n}) = 1$. Therefore, $\gamma(f, X, \hat{T}(n,r)) = \frac{1}{r+1}$. Since this example meets the lower bound shown above, it follows that $\gamma(\hat{T}(n,r)) = \frac{1}{r+1}$. The rightmost equality in (6.32) is the case where $r = r^*$.

Next we show the middle inequality in 6.32: $\gamma(G^*_{m,n}) \geq \gamma(\hat{T}(n,r^*))$. We claim that

$$r^* = \min_{\hat{T}(n,r):\hat{m}\leq m} r, \tag{6.39}$$

where $\hat{m}$ is the number of edges in $\hat{T}(n,r)$. The equations (6.37) and (6.35) imply that in order to guarantee $\hat{T}(n,r)$ does not have more than m edges the following must be true:

$$\frac{1}{2}n(n-1) - m \leq \left(1-\frac{1}{r}\right)\frac{n^2}{2}, \tag{6.40}$$

since $\bar{m} = \frac{1}{2}n(n-1) - m$. With some algebraic manipulation, this implies

$$\frac{n^2}{2m+n} \leq r. \tag{6.41}$$

Since r must be a positive integer, the lowest value of r is r^* as defined in (6.33), therefore (6.39) holds.

Let $G_{m,n} \in \mathscr{G}_{m,n}$ be a graph which is created by starting with $\hat{T}(n,r^*)$ and adding $m-\hat{m}$ edges. Since adding edges cannot remove any cliques, $k(G_{m,n}) \leq k(\hat{T}(n,r^*)) = r^*$. Thus by Theorem 11

$$\gamma(G^*_{m,n}) \geq \gamma(G_{m,n}) \geq \frac{1}{k(G_{m,n})+1} \geq \frac{1}{r^*+1}, \tag{6.42}$$

proving the middle inequality.

Finally, we show the leftmost inequality $\frac{1}{r^*} \geq \gamma(G^*_{m,n})$. According to (6.38) and (6.39), one cannot achieve a lower independence number than r^* in a graph with n vertices and m edges. Therefore, by Theorem 11 this inequality holds. ■

6.5 Conclusion

In this chapter, we have shown bounds on the worst-case efficiency of the distributed greedy algorithm for submodular maximization. These bounds show how to design communication structures that maximize the worst-case efficiency.

Future research can follow in several directions. For example, while the bounds presented in Section III are applicable to any graph structure $G \in \mathscr{G}$, it is not clear how to characterize graphs where $\gamma(G) = \frac{1}{\alpha(G)}$ versus those where $\gamma(G) = \frac{1}{k(G)+1}$. It is also not clear whether there exists any graph such that $\frac{1}{\alpha(G)} > \gamma(G) > \frac{1}{k(G)+1}$. Precisely characterizing $\gamma(G)$ for any graph structure will also lead to an exact formulation on how to construct any $G^*_{m,n}$.

Another idea for future research is to consider a game-theoretic approach instead of a greedy approach, similar to work done in [117]. In this case, the order of the agents wouldn't matter, and the goal would be to define characteristics of the Nash equilibrium for certain graph structures. Finally, future work could include exploring the use of a different utility function rather than marginal contribution in order to make greedy decisions. The key point of study here would be whether a different set of functions could yield better performance guarantees than marginal contribution.

Conclusion

The goal of the thesis is to analyze distributed learning algorithms and to apply them to wireless communication networks. We study two algorithms for near-potential games: Log-Linear Learning Algorithm (LLA) and Binary LLA (BLLA) for full and partial information settings respectively. We propose new rules for computation of resistance of trees of a Perturbed Markov Chain (PMC). These rules enable us to prove the convergence of the algorithms. We prove that the algorithms converge to the global minimum of the potential function under a certain condition, which captures the tradeoff between the quality of the obtained solution and the distance to an exact potential game. Next, we analyze the dynamics of BLLA in noisy-potential games, which take into account noisy utilities. We prove the convergence of BLLA using the proposed rules of resistance computation for fixed temperature and decreasing temperature parameter. With fixed temperature τ, the convergence in probability is obtained while with decreasing temperature $\tau(t)$ an almost sure convergence is obtained. We identify the sufficient number of estimation samples to be used that guarantees the convergence to the global maximum of the potential function. We prove the convergence of BLLA for two cases of bounded and unbounded noise. In bounded noise case, the number of samples depends on the range of noise. In unbounded noise case, the number of samples is a function of the moment generating function of the noise distribution.

We apply LLA and BLLA in near-potential games for load balancing among the base stations in a heterogeneous cellular networks using Cell Range Extension (CRE) bias for user association and Almost Blank Subframe (ABS) for interference management. We consider an α-fairness objective function with outage and load constraints. The α-fairness function captures various network performance of the network. We obtain rate optimal policy or minimum sum load policy for $\alpha = 0$, proportional fairness for $\alpha = 1$, minimum average delay of the network for $\alpha = 2$, and min-max load policy as $\alpha \to \infty$. We derive practical conditions in terms of the system parameter that guarantee the convergence of LLA and BLLA. We perform extensive simulations that show that the proposed algorithms

achieve the desired load balancing. It is observed that near-potential game model naturally arises under shadow fading scenario. We show that ABS helps to meet the outage constraint and to achieve a better load balancing.

We also develop a new Annealing Learning Algorithm (ALA) for automatically finding the best temperature τ to be used for LLA and BLLA. We show through simulations that ALA performs better than other usually used annealing schedules in the literature.

Next, we apply BLLA in noisy-potential games to Channel Assignment Problem (CAP) in Device-to-Device (D2D) wireless networks, when throughput estimation is corrupted by noise. By using the sufficient number of samples from the convergence results of BLLA the optimal channel assignment is achieved. We observe that BLLA achieves the maximum sum data rate of the network. The sum data rate increases with the number of channels and with the number of users. BLLA performs better than better response algorithm in the presence of noise.

Finally, we address a general constrained submodular maximization problem using games with players having limited or no information about their neighborhood. We give lower and upper bounds on the worst-case performance of the greedy algorithm for a given information graph. Then, we determine the best information graph that gives the best performance that a system designer can achieve with a fixed number of agents and information edges. These results show that when information is costly, the best communication graphs spread out the communication links among the agents, rather than clustering them among a small group.

Bibliography

[1] J. Mo and J. Walrand, "Fair end-to-end window-based congestion control," *IEEE/ACM Trans. Netw.*, vol. 8, no. 5, pp. 556–567, 2000.

[2] J.-Y. Audibert and S. Bubeck, "Best arm identification in multi-armed bandits," in *COLT-23th Conference on Learning Theory*, pp. 13–p, 2010.

[3] D. Monderer and L. S. Shapley, "Potential Games," *Games and Economic Behavior*, vol. 14, pp. 124–143, May 1996.

[4] G. Como, F. Fagnani, and S. Zampieri, "Distributed learning in potential games over large-scale networks," in *The 21st International Symposium on Mathematical Theory of Networks and Systems (MTNS 2014)*, 2014.

[5] O. Candogan, A. Ozdaglar, and P. A. Parrilo, "Learning in near-potential games," in *IEEE Conf. Decision Control and European Control Conference*, pp. 2428–2433, 2011.

[6] M. S. Ali, P. Coucheney, and M. Coupechoux, "Load balancing in heterogeneous networks based on distributed learning in near-potential games," *IEEE Transactions on Wireless Communications*, vol. 15, no. 7, pp. 5046–5059, 2016.

[7] D. S. Leslie and J. R. Marden, "Equilibrium selection in potential games with noisy rewards," in *Proc. IEEE Network Games, Control and Optimization (NetGCooP)*, pp. 1–4, 2011.

[8] M. S. Ali, P. Coucheney, and M. Coupechoux, "Optimal distributed channel assignment in D2D networks using learning in noisy potential games," *arXiv preprint arXiv:1701.04577*, 2017.

[9] H. P. Young, "The Evolution of Conventions," *Econometrica*, vol. 61, pp. 57–84, Jan. 1993.

[10] D. Monderer and L. S. Shapley, "Fictitious play property for games with identical interests," *Journal of economic theory*, vol. 68, no. 1, pp. 258–265, 1996.

[11] C. Alós-Ferrer and N. Netzer, "The logit-response dynamics," *Games and Economic Behavior*, vol. 68, no. 2, pp. 413–427, 2010.

[12] L. E. Blume, "The statistical mechanics of strategic interaction," *Games and economic behavior*, vol. 5, no. 3, pp. 387–424, 1993.

[13] L. E. Blume *et al.*, "Population games," Santa Fe Institute, 1995.

[14] H. P. Young, *Individual strategy and social structure: An evolutionary theory of institutions*. Princeton University Press, 2001.

[15] J. R. Marden, G. Arslan, and J. S. Shamma, "Connections between cooperative control and potential games illustrated on the consensus problem," in *Control Conference (ECC), 2007 European*, pp. 4604–4611, IEEE, 2007.

[16] L. E. Blume, "How noise matters," *Games and Economic Behavior*, vol. 44, no. 2, pp. 251–271, 2003.

[17] G. Arslan, J. R. Marden, and J. S. Shamma, "Autonomous vehicle-target assignment: A game-theoretical formulation," *Journal of Dynamic Systems, Measurement, and Control*, vol. 129, no. 5, pp. 584–596, 2007.

[18] D. Shah and J. Shin, "Dynamics in congestion games," in *ACM SIGMETRICS Performance Evaluation Review*, vol. 38, pp. 107–118, ACM, 2010.

[19] M. Benaim and W. Sandholm, "Logit evolution in potential games: Reversibility, rates of convergence, large deviations, and equilibrium selection," *Unpublished manuscript, Université de Neuchâtel and University of Wisconsin*, 2007.

[20] J. R. Marden and J. S. Shamma, "Revisiting log-linear learning: Asynchrony, completeness and a payoff-based implementation," *Games and Economic Behaviour*, vol. 75, pp. 788–808, July 2012.

[21] B. S. Pradelski and H. P. Young, "Learning efficient Nash equilibria in distributed systems," *Games and Economic Behavior*, vol. 75, pp. 882–897, July 2012.

[22] H. P. Young, "Learning by Trial and Error," *Games and Economic Behavior*, vol. 65, no. 2, pp. 626–643, 2009.

[23] N. Li and J. R. Marden, "Designing games for distributed optimization," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 2, pp. 230–242, 2013.

[24] Y. Xu, J. Wang, Q. Wu, A. Anpalagan, and Y.-D. Yao, "Opportunistic spectrum access in cognitive radio networks: Global optimization using local interaction games," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 6, pp. 180–194, April 2012.

[25] K. Yamamoto, "A comprehensive survey of potential game approaches to wireless networks," *IEICE Transactions on Communications*, vol. 98, no. 9, pp. 1804–1823, 2015.

[26] B. Hajek, "Cooling schedules for optimal annealing," *Mathematics of operations research*, vol. 13, no. 2, pp. 311–329, 1988.

[27] S. Anily and A. Federgruen, "Ergodicity in parametric nonstationary markov chains: An application to simulated annealing methods," *Operations Research*, vol. 35, no. 6, pp. 867–874, 1987.

[28] S. Anily and A. Federgruen, "Simulated annealing methods with general acceptance probabilities," *Journal of Applied Probability*, pp. 657–667, 1987.

[29] P. Brémaud, *Markov chains: Gibbs fields, Monte Carlo simulation, and queues*, vol. 31. Springer Science & Business Media, 2013.

[30] V. Anantharam and P. Tsoucas, "A proof of the markov chain tree theorem," *Statistics & Probability Letters*, vol. 8, no. 2, pp. 189–192, 1989.

[31] J. Andrews, S. Singh, Q. Ye, X. Lin, and H. Dhillon, "An overview of load balancing in hetnets: old myths and open problems," *IEEE Wireless Commun.*, vol. 21, pp. 18–25, April 2014.

[32] O. K. Tonguz and E. Yanmaz, "The mathematical theory of dynamic load balancing in cellular networks," *IEEE Trans. Mobile Comput.*, vol. 7, no. 12, pp. 1504–1518, 2008.

[33] K. Son, S. Chong, and G. Veciana, "Dynamic association for load balancing and interference avoidance in multi-cell networks," *IEEE Trans. Wireless Commun.*, vol. 8, no. 7, pp. 3566–3576, 2009.

[34] B. Rengarajan and G. De Veciana, "Architecture and abstractions for environment and traffic-aware system-level coordination of wireless networks," *IEEE Trans. Netw. ACM*, vol. 19, no. 3, pp. 721–734, 2011.

[35] E. Stevens-Navarro, Y. Lin, and V. W. Wong, "An MDP-based vertical handoff decision algorithm for heterogeneous wireless networks," *IEEE Trans. Veh. Technol.*, vol. 57, no. 2, pp. 1243–1254, 2008.

[36] S.-E. Elayoubi, E. Altman, M. Haddad, and Z. Altman, "A hybrid decision approach for the association problem in heterogeneous networks," in *Proc. IEEE INFOCOM*, pp. 1–5, 2010.

[37] H. Kim, G. de Veciana, X. Yang, and M. Venkatachalam, "Distributed alpha-optimal user association and cell load balancing in wireless networks," *IEEE/ACM Trans. Netw.*, vol. 20, pp. 177–190, Feb 2012.

[38] F. Xu, C. C. Tan, Q. Li, G. Yan, and J. Wu, "Designing a practical access point association protocol," in *Proc. IEEE INFOCOM*, pp. 1–9, 2010.

[39] D. Niyato and E. Hossain, "Dynamics of network selection in heterogeneous wireless networks: An evolutionary game approach," *IEEE Trans. Veh. Technol.*, vol. 58, pp. 2008–2017, May 2009.

[40] E. Aryafar, A. Keshavarz-Haddad, M. Wang, and M. Chiang, "RAT selection games in HetNets," in *Proc. IEEE INFOCOM*, pp. 998–1006, 2013.

[41] T. Kudo and T. Ohtsuki, "Cell range expansion using distributed Q-learning in heterogeneous networks," *EURASIP J. Wireless Commun. Netw.*, pp. 1–10, Mar. 2013.

[42] K. Lee, S. Kim, S. Lee, and J. Ma, "Load balancing with transmission power control in femtocell networks," in *Advanced Communication Technology (ICACT)*, pp. 519–522, Feb 2011.

[43] 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 Commun.*, vol. 18, no. 3, pp. 10–21, 2011.

[44] J. Oh and Y. Han, "Cell selection for range expansion with almost blank subframe in heterogeneous networks," in *Proc. PIMRC*, pp. 653–657, Sept 2012.

[45] K. Kitagawa, T. Komine, T. Yamamoto, and S. Konishi, "Performance evaluation of handover in LTE-advanced systems with pico cell range expansion," in *Proc. PIMRC*, pp. 1071–1076, Sep. 2012.

[46] P. Ökvist and A. Simonsson, "LTE HetNet trial - range expansion including micro/pico indoor coverage survey," in *Proc. VTC (Fall)*, pp. 1–5, Sep 2012.

[47] S. Vasudevan, R. N. Pupala, and K. Sivanesan, "Dynamic eICIC-A proactive strategy for improving spectral efficiencies of heterogeneous LTE cellular networks by leveraging user mobility and traffic dynamics," *IEEE Trans. Wireless Commun.*, vol. 12, no. 10, pp. 4956–4969, 2013.

[48] J. Wang, J. Liu, D. Wang, J. Pang, and G. Shen, "Optimized fairness cell selection for 3GPP LTE-A macro-pico HetNets," in *Proc. VTC (Fall)*, pp. 1–5, Sep 2011.

[49] I. Siomina and D. Yuan, "Load balancing in heterogeneous LTE: Range optimization via cell offset and load-coupling characterization," in *Proc. ICC*, pp. 1357–1361, June 2012.

[50] Y. Khan, B. Sayrac, and E. Moulines, "Surrogate based centralised son: Application to interference mitigation in LTE-A hetnets," in *Proc. VTC (Spring)*, pp. 1–5, 2013.

[51] S. Deb, P. Monogioudis, J. Miernik, and J. P. Seymour, "Algorithms for enhanced inter-cell interference coordination (eICIC) in LTE hetnets," *IEEE/ACM Trans. Netw. (TON)*, vol. 22, no. 1, pp. 137–150, 2014.

[52] A. Bedekar and R. Agrawal, "Optimal muting and load balancing for eicic," in *International Symp. Modeling and Optimisation in Mobile, Ad Hoc, and Wireless Netw.*, pp. 280–287, IEEE, 2013.

[53] K. Kikuchi and H. Otsuka, "Proposal of adaptive control CRE in heterogeneous networks," in *Proc. PIMRC*, pp. 910–914, Sep 2012.

[54] M. Al-Rawi, "A dynamic approach for cell range expansion in interference coordinated LTE-advanced heterogeneous networks," in *Proc. ICCS*, pp. 533–537, Nov 2012.

[55] A. Tall, Z. Altman, and E. Altman, "Self organizing strategies for enhanced ICIC (eICIC)," in *International Symp. Modeling and Optimisation in Mobile, Ad Hoc, and Wireless Netw.*, pp. 318–325, IEEE, 2014.

[56] Q. Ye, B. Rong, Y. Chen, M. Al-Shalash, C. Caramanis, and J. Andrews, "User association for load balancing in heterogeneous cellular networks," *IEEE Trans. Wireless Commun.*, vol. 12, pp. 2706–2716, June 2013.

[57] Y. Liu, C. S. Chen, and C. W. Sung, "Joint optimization on inter-cell interference management and user attachment in lte-a hetnets," in *International Workshop on Resource Allocation, Cooperation and Competition in Wireless Networks (RAWNET)*, IEEE, 2015.

[58] S. Singh, X. Zhang, and J. Andrews, "Joint rate and sinr coverage analysis for decoupled uplink-downlink biased cell associations in hetnets," *IEEE Trans. Wireless Commun.*, vol. 14, pp. 5360–5373, Oct 2015.

[59] A. Goldsmith, *Wireless communications*. Cambridge university press, 2005.

[60] M. Gudmundson, "Correlation model for shadow fading in mobile radio systems," *Electron. Lett.*, vol. 27, no. 23, pp. 2145–2146, 1991.

[61] R. Fraile, J. F. Monserrat, J. Gozálvez, and N. Cardona, "Mobile radio bi-dimensional large-scale fading modelling with site-to-site cross-correlation," *European trans. on telecommunications*, vol. 19, no. 1, pp. 101–106, 2008.

[62] R. Madan, J. Borran, A. Sampath, N. Bhushan, A. Khandekar, and T. Ji, "Cell association and interference coordination in heterogeneous LTE-A cellular networks," *IEEE J. Sel. Areas Commun.*, vol. 28, pp. 1479–1489, Dec. 2010.

[63] K. Okino, T. Nakayama, C. Yamazaki, H. Sato, and Y. Kusano, "Pico cell range expansion with interference mitigation toward lte-advanced heterogeneous networks," in *ICC Workshop*, pp. 1–5, June 2011.

[64] M. Shirakabe, A. Morimoto, and N. Miki, "Performance evaluation of inter-cell interference coordination and cell range expansion in heterogeneous networks for LTE-advanced downlink," in *ISWCS*, pp. 844–848, Nov. 2011.

[65] M. Vajapeyam, A. Damnjanovic, J. Montojo, T. Ji, Y. Wei, and D. Malladi, "Downlink FTP performance of heterogeneous networks for LTE-advanced," in *ICC Workshop*, pp. 1–5, June 2011.

[66] I. Guvenc, "Capacity and fairness analysis of heterogeneous networks with range expansion and interference coordination," *IEEE Commun. Lett.*, vol. 15, pp. 1084–1087, Oct. 2011.

[67] A. Morimoto, N. Miki, H. Ishii, and D. Nishikawa, "Investigation on transmission power control in heterogeneous network employing cell range expansion for LTE-advanced uplink," in *European Wireless Conf.*, pp. 1–6, Apr. 2012.

[68] M. Eguizabal and A. Hernandez, "Interference management and cell range expansion analysis for LTE picocell deployments," in *Proc. PIMRC*, pp. 1592–1597, Sept 2013.

[69] T. Bonald and L. Massoulié, "Impact of fairness on internet performance," in *ACM SIGMETRICS Performance Evaluation Review*, vol. 29, pp. 82–91, 2001.

[70] 3GPP, "Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access (E-UTRAN); Overall description; Stage 2," TS 36.300, 3GPP, Sept. 2008.

[71] 3GPP, "Technical specification group radio access network; evolved universal terrestrial radio access (eutra); further advancements for e-utra physical layer aspects," TR 36.814, 3GPP, Mar. 2010.

[72] B. S. Pradelski and H. P. Young, "Learning efficient nash equilibria in distributed systems," *Games and Economic behavior*, vol. 75, no. 2, pp. 882–897, 2012.

[73] H. Borowski and J. Marden, "Fast convergence in semi-anonymous potential games," *IEEE Transactions on Control of Network Systems*, 2015.

[74] Y. Nourani and B. Andresen, "A comparison of simulated annealing cooling strategies," *Journal of Physics A: Mathematical and General*, vol. 31, no. 41, p. 8373, 1998.

[75] M. S. Ali, P. Coucheney, and M. Coupechoux, "Load balancing in heterogeneous networks based on distributed learning in potential games," in *International Symp. Modeling and Optimisation in Mobile, Ad Hoc, and Wireless Netw.*, 2015.

[76] A. Asadi, Q. Wang, and V. Mancuso, "A survey on device-to-device communication in cellular networks," *IEEE Commun. Surveys & Tutorials*, vol. 16, no. 4, pp. 1801–1819, 2014.

[77] M. N. Tehrani, M. Uysal, and H. Yanikomeroglu, "Device-to-device communication in 5g cellular networks: challenges, solutions, and future directions," *IEEE Commun. Mag.*, vol. 52, no. 5, pp. 86–92, 2014.

[78] L. Song, D. Niyato, Z. Han, and E. Hossain, "Game-theoretic resource allocation methods for device-to-device communication," *IEEE Commun. Lett.*, vol. 21, no. 3, pp. 136–144, 2014.

[79] Q. Ye, M. Al-Shalash, C. Caramanis, and J. G. Andrews, "Distributed resource allocation in device-to-device enhanced cellular networks," *IEEE Trans. Commun.*, vol. 63, no. 2, pp. 441–454, 2015.

[80] Y. Li, D. Jin, J. Yuan, and Z. Han, "Coalitional games for resource allocation in the device-to-device uplink underlaying cellular networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 7, pp. 3965–3977, 2014.

[81] H. Chen, D. Wu, and Y. Cai, "Coalition formation game for green resource management in d2d communications," *IEEE Commun. Lett.*, vol. 18, no. 8, pp. 1395–1398, 2014.

[82] Y. Cai, H. Chen, D. Wu, W. Yang, and L. Zhou, "A distributed resource management scheme for d2d communications based on coalition formation game," in *Proc. ICC*, pp. 355–359, IEEE, 2014.

[83] B.-Y. Huang, S.-T. Su, C.-Y. Wang, C.-W. Yeh, and H.-Y. Wei, "Resource allocation in d2d communication-a game theoretic approach," in *Proc. ICC*, pp. 483–488, IEEE, 2014.

[84] C. Xu, L. Song, Z. Han, Q. Zhao, X. Wang, X. Cheng, and B. Jiao, "Efficiency resource allocation for device-to-device underlay communication systems: a reverse iterative combinatorial auction based approach," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 9, pp. 348–358, 2013.

[85] R. Wang, J. Zhang, S. Song, and K. B. Letaief, "Optimal qos-aware channel assignment in d2d communications with partial csi," *IEEE Trans. Wireless Commun.*, vol. 15, no. 11, pp. 7594–7609, 2016.

[86] S. Maghsudi and S. Stańczak, "Hybrid centralized–distributed resource allocation for device-to-device communication underlaying cellular networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 4, pp. 2481–2495, 2016.

[87] C. Gao, X. Sheng, J. Tang, W. Zhang, S. Zou, and M. Guizani, "Joint mode selection, channel allocation and power assignment for green device-to-device communications," in *Proc. ICC*, pp. 178–183, 2014.

[88] M. R. Garey and D. S. Johnson, *Computers and intractability*, vol. 29. wh freeman New York, 2002.

[89] E. Ahmed, A. Gani, S. Abolfazli, L. J. Yao, and S. U. Khan, "Channel assignment algorithms in cognitive radio networks: Taxonomy, open issues, and challenges," *IEEE Commun. Surveys & Tutorials*, vol. 18, no. 1, pp. 795–823, 2016.

[90] G. K. Audhya, K. Sinha, S. C. Ghosh, and B. P. Sinha, "A survey on the channel assignment problem in wireless networks," *Wireless Commun. and Mobile Computing*, vol. 11, no. 5, pp. 583–609, 2011.

[91] P. T. Chan, M. Palaniswami, and D. Everitt, "Neural network-based dynamic channel assignment for cellular mobile communication systems," *IEEE Trans. Veh. Technol.*, vol. 43, no. 2, pp. 279–288, 1994.

[92] M. Duque-Antón, D. Kunz, and B. Rüber, "Channel assignment for cellular radio using simulated annealing," *IEEE Trans. Veh. Technol.*, vol. 42, no. 1, pp. 14–21, 1993.

[93] J. C. Spall, *Introduction to stochastic search and optimization: estimation, simulation, and control*, vol. 65. John Wiley & Sons, 2005.

[94] J. Chen, Q. Yu, P. Cheng, Y. Sun, Y. Fan, and X. Shen, "Game theoretical approach for channel allocation in wireless sensor and actuator networks," *IEEE Trans. Automatic Control*, vol. 56, no. 10, pp. 2332–2344, 2011.

[95] A. Krause and C. Guestrin, "Near-optimal observation selection using submodular functions," in *AAAI*, vol. 7, pp. 1650–1654, 2007.

[96] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 137–146, ACM, 2003.

[97] P. Kohli, M. P. Kumar, and P. H. Torr, "P^3 & beyond: Move making algorithms for solving higher order functions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 9, pp. 1645–1656, 2009.

[98] O. Barinova, V. Lempitsky, and P. Kholi, "On detection of multiple object instances using hough transforms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 9, pp. 1773–1784, 2012.

[99] H. Lin and J. Bilmes, "A class of submodular functions for document summarization," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pp. 510–520, Association for Computational Linguistics, 2011.

[100] A. Singh, A. Krause, C. Guestrin, W. J. Kaiser, and M. A. Batalin, "Efficient planning of informative paths for multiple robots.," in *IJCAI*, vol. 7, pp. 2204–2211, 2007.

[101] A. Krause, R. Rajagopal, A. Gupta, and C. Guestrin, "Simultaneous placement and scheduling of sensors," in *Proceedings of the 2009 international Conference on information Processing in Sensor Networks*, pp. 181–192, IEEE Computer Society, 2009.

[102] J. R. Marden, "The role of information in distributed resource allocation," *IEEE Transactions on Control of Network Systems*, 2016.

[103] M. Grötschel, L. Lovász, and A. Schrijver, "The ellipsoid method and its consequences in combinatorial optimization," *Combinatorica*, vol. 1, no. 2, pp. 169–197, 1981.

[104] S. Iwata, L. Fleischer, and S. Fujishige, "A combinatorial strongly polynomial algorithm for minimizing submodular functions," *Journal of the ACM (JACM)*, vol. 48, no. 4, pp. 761–777, 2001.

[105] A. Schrijver, "A combinatorial algorithm minimizing submodular functions in strongly polynomial time," *Journal of Combinatorial Theory, Series B*, vol. 80, no. 2, pp. 346–355, 2000.

[106] L. Lovász, "Submodular functions and convexity," in *Mathematical Programming The State of the Art*, pp. 235–257, Springer, 1983.

[107] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, "An analysis of approximations for maximizing submodular set functions I," *Mathematical Programming*, vol. 14, no. 1, pp. 265–294, 1978.

[108] M. L. Fisher, G. L. Nemhauser, and L. A. Wolsey, "An analysis of approximations for maximizing submodular set functions II," in *Polyhedral combinatorics*, pp. 73–87, Springer, 1978.

[109] M. Minoux, "Accelerated greedy algorithms for maximizing submodular set functions," in *Optimization Techniques*, pp. 234–243, Springer, 1978.

[110] N. Buchbinder, M. Feldman, J. Seffi, and R. Schwartz, "A tight linear time (1/2)-approximation for unconstrained submodular maximization," *SIAM Journal on Computing*, vol. 44, no. 5, pp. 1384–1402, 2015.

[111] J. Vondrák, "Optimal approximation for the submodular welfare problem in the value oracle model," in *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pp. 67–74, ACM, 2008.

[112] M. Sviridenko, "A note on maximizing a submodular set function subject to a knapsack constraint," *Operations Research Letters*, vol. 32, no. 1, pp. 41–43, 2004.

[113] G. Qu, D. Brown, and N. Li, "Distributed greedy algorithm for mulit-agent task assignment problem with submodular utility functions." Preprint submitted to Automatica, December 2016.

[114] J. R. Marden, "The role of information in multiagent coordination," in *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*, pp. 445–450, IEEE, 2014.

[115] B. Gharesifard and S. L. Smith, "On distributed submodular maximization with limited information," in *American Control Conference (ACC)*, pp. 1048–1053, IEEE, 2016.

[116] P. Turán, "On an extremal problem in graph theory," *Mat. Fiz. Lapok*, vol. 48, no. 436-452, p. 137, 1941.

[117] A. Vetta, "Nash equilibria in competitive societies, with applications to facility location, traffic routing and auctions," in *Foundations of Computer Science, 2002. Proceedings. The 43rd Annual IEEE Symposium on*, pp. 416–425, IEEE, 2002.