Springer Theses
Recognizing Outstanding Ph.D. Research

Seyed Rasoul Etesami

Potential-Based Analysis of Social, Communication, and Distributed Networks

Springer

Springer Theses

Recognizing Outstanding Ph.D. Research

Aims and Scope

The series “Springer Theses” brings together a selection of the very best Ph.D. theses from around the world and across the physical sciences. Nominated and endorsed by two recognized specialists, each published volume has been selected for its scientific excellence and the high impact of its contents for the pertinent field of research. For greater accessibility to non-specialists, the published versions include an extended introduction, as well as a foreword by the student’s supervisor explaining the special relevance of the work for the field. As a whole, the series will provide a valuable resource both for newcomers to the research fields described, and for other scientists seeking detailed background information on special questions. Finally, it provides an accredited documentation of the valuable contributions made by today’s younger generation of scientists.

Theses are accepted into the series by invited nomination only and must fulfill all of the following criteria

- They must be written in good English.
- The topic should fall within the confines of Chemistry, Physics, Earth Sciences, Engineering and related interdisciplinary fields such as Materials, Nanoscience, Chemical Engineering, Complex Systems and Biophysics.
- The work reported in the thesis must represent a significant scientific advance.
- If the thesis includes previously published material, permission to reproduce this must be gained from the respective copyright holder.
- They must have been examined and passed during the 12 months prior to nomination.
- Each thesis should include a foreword by the supervisor outlining the significance of its content.
- The theses should have a clearly defined structure including an introduction accessible to scientists not expert in that particular field.

More information about this series at http://www.springer.com/series/8790

Seyed Rasoul Etesami

Potential-Based Analysis of Social, Communication, and Distributed Networks

Doctoral Thesis accepted by
University of Illinois at Urbana–Champaign, Champaign,
USA

Springer

Author
Dr. Seyed Rasoul Etesami
Department of Electrical Engineering
Princeton University
Princeton, NJ
USA

Supervisor
Prof. Tamer Başar
Department of Electrical and Computer Engineering
University of Illinois at Urbana–Champaign
Champaign, IL
USA

ISSN 2190-5053 ISSN 2190-5061 (electronic)
Springer Theses
ISBN 978-3-319-54288-1 ISBN 978-3-319-54289-8 (eBook)
DOI 10.1007/978-3-319-54289-8

Library of Congress Control Number: 2017932772

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

In the name of God
To my family, for their love and support.

Supervisor's Foreword

In recent years, there has been a wide range of studies on the role of social and distributed networks. In particular, availability of large amounts of data from online social networks and advances in control of distributed systems have drawn the attention of many researchers to exploit the connections among several evolutionary behaviors in social, communication and distributed networks. The Ph.D. thesis of Rasoul Etesami makes major contributions to this thriving area, by introducing novel methodologies and tools toward study of evolutionary behavior of social, communication, and distributed networks, as well as their computational complexity and rates of convergence. The thesis provides a collection of new results related to four different types of problems that arise in this general context. The study undertaken and the approaches adopted enable the analysis of the evolution of certain types of social and distributed networks, and also identification of local and global patterns of their dynamics using some novel potential-theoretic techniques.

The thesis covers four main topics, and in describing its main contributions, this *Foreword* has been organized along similar lines:

1. **Quantized Consensus.** In this part of the thesis, the focus is on analysis of a specific type of distributed algorithms, known as *unbiased quantized* algorithm, where a set of agents, handling only integer-valued data, interact locally in a network in order to reach a consensus. The thesis provides tight expressions for the expected convergence time of the emerging dynamics over general static and time-varying networks. In addition, the thesis introduces new protocols using *Metropolis* chains and obtains the fastest (as of today) randomized quantized consensus protocol over time-varying networks, which substantially improves the state of the art. Two of the main novelties of this part:

   - Introducing a new technique for decoupling random walks using harmonic functions.
   - Bounding the expected convergence time of the quantized consensus protocol by bounding the spectral gap of the transition probabilities of the underlying dynamics.

2. **Opinion Dynamics.** On this topic, the thesis analyzes a class of distributed averaging dynamics, known as *Hegselmann–Krause opinion dynamics*, by focusing on their evolutionary properties. In particular, it answers an open question related to the termination time of such dynamics by establishing a connection between the convergence time and the spectral gap of the adjacency matrices of the underlying dynamics. This novel approach not only brings an improvement on the best-known upper bound (until this study), but also removes the dependency of termination time from the dimension of the ambient space. The thesis also obtains the first polynomial bounds on the termination time of the asynchronous version of such dynamics. Two of the main novelties of this part:

   - Building a bridge between two classes of problems, namely distributed control problems and game problems.
   - Establishing a connection between the rate of increase of the so-called *kinetic-s-energy* associated with multi-agent systems and the spectral gap of their underlying dynamics.

3. **Resource Allocation.** In this part, the thesis considers a class of resource allocation games over networks and devises some easily implementable distributed algorithms which drive the entire network to a *Nash equilibrium* in polynomial time for dense and hierarchical networks. In particular, it is shown that such games benefit from having low price of anarchy (PoA), and hence can be used to model allocation systems which suffer from lack of coordination. This discovery enables Rasoul to devise a distributed approximation algorithm within a constant factor of any pure-strategy Nash equilibrium over general networks. Two of the main novelties of this part of the thesis:

   - Novel use of probabilistic and combinatorial approaches to bound PoA.
   - Using a boosting technique to construct a new potential function in order to approximate the equilibrium points of the system.

4. **Diffusion Dynamics.** On this topic, the thesis studies an important problem related to competition over social networks. It establishes a hardness result for searching an equilibrium over a class of games known as *competitive diffusion* games, and provides some necessary conditions for the existence of a pure-strategy Nash equilibrium. In particular, the thesis provides some *concentration* results related to the expected utility of the players over random graphs. Two of the main novelties of this part:

   - Making a connection between equilibrium points of the diffusion game and solutions of an NP-hard combinatorial problem.
   - Novel use of martingale techniques to analyze the diffusion game over random networks.

In terms of overall impact, the thesis provides several easily implementable algorithms which deliver excellent performance guarantees while running faster than those that exist in the literature. By departing from the classic approaches and results in the literature, and by introducing new approaches and tools, the thesis shows that it is possible to handle more complex and realistic nonlinear models where the traditional approaches either fail or lead to weak results. These new techniques developed in the thesis have the potential to apply to and resolve other outstanding issues in social and distributed networks of similar types.

Champaign, USA Prof. Tamer Başar
January 2017

Preface

Social, communication, and distributed networks spread throughout our daily lives. They play an important role in information transmission with others and are critical for trade of goods and services. They explain how we vote, how we choose our friends, or even how we select our language. They determine how disease spreads through human societies, or what products individuals tend to buy in a competitive market [1]. With the appearance of myriads of large-scale social networks and availability of huge data sets, modeling and analysis of social networks under a distributed setting have gained a lot of attention in the research community in recent years. This is partly because social networks are usually made as a collection of self-interested individuals who seek their own objectives, and partly because analyzing such networks as a whole using a central identity is either computationally expensive or not feasible. In particular, recent advances in control of distributed systems and their effectiveness in the analysis of large-scale complex networks have drawn the attention of many researchers to exploit the connections among evolutionary behaviors in social and distributed networks [2].

The objective of this thesis is to provide a foundation for analyzing and understanding social and distributed networks using new methodologies and techniques. We first revisit several well-known types of social and distributed networks and review some relevant results from the literature. Building on this, we present a set of new results related to four different types of problems, and identify several directions for future research. The study undertaken and the approaches adopted allow us to analyze the evolution and behavior of certain types of networked decision systems and also to identify local and global patterns of their dynamics using some novel potential-theoretic techniques.

Following the introduction and preliminaries, we first focus on analyzing a specific type of distributed algorithms known as an *unbiased quantized consensus* algorithm where a set of agents interact locally in a network in order to reach a consensus. Such protocols are especially useful when there is limited memory and limited communication energy. It is well understood by now that protocols for consensus play an important role in a number of more sophisticated multi-agent tasks such as distributed optimization, formation control, cooperative statistical inference, robotic rendezvous, power control, and load balancing, among many others, with proposed solutions relying crucially on consensus. Therefore, any improvement in design and analysis of consensus protocols can lead to faster and more efficient algorithms for many practical applications. In this regard we analyze the expected convergence time of such unbiased quantized dynamics and provide tight expressions for their convergence time over general static and time-varying networks. Following this, we introduce new protocols using a special class of Markov chains known as *Metropolis* chains and obtain the fastest (as of today) randomized quantized consensus protocol. The bounds provided here considerably improve the state of the art over static and dynamic networks.

We make a bridge between two classes of problems, namely averaging opinion dynamics and game problems [3]. Opinion dynamics in social networks is an important area of research with a wide range of applications in psychology, economics, political science, and electrical engineering. A natural question that commonly arises in this context is the extent to which one can predict the outcome of the opinion formation of entities under some complex interaction process running among these social actors. In this regard, we analyze a class of distributed averaging dynamics known as Hegselmann–Krause opinion dynamics. Modeling such dynamics as a non-cooperative game problem, we elaborate on some of the evolutionary properties of its dynamics. In particular, we answer an open question related to the termination time of such dynamics by connecting the convergence time to the spectral gap of the adjacency matrices of the underlying dynamics. This not only allows us to improve the best-known upper bound on time, but also provides a technique to connect the rate of increase of a so-called *total-s-energy* associated with multi-agent systems to the spectral gap of their underlying dynamics.

In the third part of the thesis we describe a richer class of distributed systems where the agents involved in the network act in a more strategic manner. In fact, one of the main challenges of studying social networks from a distributed control perspective is modeling the competitive behavior of social entities. In this context, game theory has been proven to be an effective and powerful tool for modeling and analyzing competitions across networks [3]. More specifically, we consider a class of resource allocation games where a set of agents compete for the same set of resources over the network. Analyzing such allocation systems are especially important for goods allocation in shopping networks, data storage in computer networks, content management, as well as web caches in distributed systems. We study the evolution of resource allocation games and processes to some final outcomes such as Nash equilibrium, and devise some simple distributed algorithms which drive the entire network to a Nash equilibrium in polynomial time for dense and hierarchical networks. In particular, we show that such games benefit from having low price of anarchy, and hence can be used to model allocation systems which suffer from lack of coordination. This fact allows us to devise a distributed approximation algorithm within a constant gap of any pure-strategy Nash equilibrium over general networks.

Subsequently we turn our attention to an important problem related to competition over social networks where the goal is to propagate a certain type of product or behavior in a desired way through a network. As an example, one can consider competition between cell phone companies, where each of them is trying to attract the largest share of customers to its own products. As diffusion models are being widely used to study competition and rational decision-making in social and economic networks, in the last part of the thesis we consider a class of games known as *competitive diffusion* games and study the connection between the social network structure and the decision-making process of the players. We establish a hardness result for searching for an equilibrium in competitive diffusion games, meaning that finding Nash equilibrium over such networks is computationally hard in a strong sense, and provide some necessary conditions for the existence of a pure-strategy Nash equilibrium in such social games. In particular, we study the behavior of the diffusion game over random graphs and provide some concentration results related to the expected utility of the players, which guarantees certain payoff levels for players as the size of the network grows larger.

Finally, we discuss some future research directions by identifying several interesting and promising problems and justify their importance in this general topical area. Several auxiliary lemmas and other complementary results are provided in Appendices at the end of the thesis.

Princeton, USA Seyed Rasoul Etesami

References

1. M. O. Jackson, Social and Economic Networks. Princeton University Press, 2010.
2. F. Bullo, J. Cortes, and S. Martinez, Distributed Control of Robotic Networks. Princeton University Press, 2009.
3. T. Başar and G. J. Olsder, Dynamic Noncooperative Game Theory. SIAM, 1999, vol. 23.

Acknowledgements

First of all, I would like to express my deepest gratitude to my thesis adviser, Prof. Tamer Başar, for his guidance, his patience as well as the freedom he gave me during my graduate studies. I had the honor to complete my Ph.D. under his supervision. During this time I found him deep in knowledge, humble in attitude, firm in reasoning, and patient in action. Working with him was always enjoyable to me because he would give me enough time to learn and explore different areas and keep his confidence in my scholarly abilities. This thesis could not have been completed without his support, and I will consider myself always in his debt for other numerous reasons which cannot be expressed in this limited space.

Furthermore, I would like to thank my thesis committee members: Prof. Bruce Hajek, for kindly accepting to be among my committee members and for his important comments and constructive ideas during the research process. Professor Rayadurgam Srikant, for his helpful comments and his friendly personality. It was my pleasure to serve as a teaching assistant for his game theory course in which I learned so much during his lectures. Many thanks to Prof. Angelia Nedić , not only as my committee member, but also as my master's thesis advisor. She was a great mentor and extremely helpful person, and I owe her for all of her support. I am especially thankful to Prof. Alex Olshevsky for the content of Chap. 4 which was carried out as a joint work during his fabulous distributed control course, and had a great impact on the contents of this thesis.

Also, I would like to thank my wonderful friend, Behrouz Touri, for all his help. Talking to him was always beneficial to me, as I could learn so many new things from him, but not necessarily vice versa! In addition, I would like to thank my awesome officemates: Jun Moon, for having great but useless discussions, Xiaobin Gao, for bringing energy to the group, Khaled Alshehri, Ali Khanafer, Abhishek Gupta, Kamil Nar, Zhi Xu, Bahman Gharesifard, Sadegh Bolouki, Emrah Akyol, Xudong Chen, Ji Liu, and Mohammad Sayin. Also many thanks to my friends Adel Ahmadiyan, Seyed Mohammad Nourbakhsh, and Ehsan Totoni for having exotic game nights in Ostad's home!

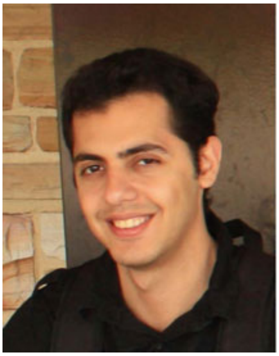
I would like to express my deepest thanks to my wonderful father. Even though I felt his warmth very briefly in my life, his memory will always remain in my heart, and I am sure he would be glad to see this process through to its completion. After that, my special thanks go to my mom for tolerating all the pressures of raising me, and sacrificing her rights by allowing me to study abroad right at the time when she needed me the most. Also, I would like to thank my family members: My brother Kamal for his help and support. Many thanks to my dear sisters, Parvin, Nasrin, and Mehri for their encouragement and intellectual support, as I would not be here without any of them. In particular, I want to thank my brother, Jalal, as we were influenced by each other in most aspects of our lives and grew up together like twins.

Last but not least, I want to offer my special thanks to my wife, Xing Gao, for her love and support. She would always bring happiness to me and tolerate my mistakes with her kindness. She sacrificed so much by giving up her better career opportunities to accompany me, stood next to me with all the difficulties, and gave me motivation and energy until the completion of this thesis.

This research was supported in its various stages by the "Cognitive & Algorithmic Decision Making" project grant through the College of Engineering of the University of Illinois, and in part by AFOSR MURI Grant FA 9550-10-1-0573 and NSF grant CCF 11-11342. They are very much appreciated.

Contents

About the Author



Dr. Seyed Rasoul Etesami (S'12-M'16) received his B.S. degree in Electrical Engineering from Isfahan University of Technology, Isfahan, Iran in 2010, M.S. degree in Industrial and Enterprise Systems Engineering in 2012, M.S. degree in Applied Mathematics in 2015, and Ph.D. degree in Electrical and Computer Engineering, all from University of Illinois at Urbana–Champaign. Currently, he is a postdoctoral research fellow with the Department of Electrical Engineering at Princeton University where he is working on developing new models for design and analysis of smart grids. His research interests include social and distributed networks, networked games, smart grids and resource allocation, algorithm design, and computational complexity. He received the best CSL Ph.D. Thesis Award at the Engineering College of the University of Illinois at Urbana–Champaign in 2016.

Symbols

$\mathbb{N}$	The set of positive integer numbers
$\mathbb{Z}$	The set of positive integer numbers
$\mathbb{R}$	The set of real numbers
$\\|\cdot\\|$	The Euclidean norm of a vector
$\oplus$	The binary sum operation
$conv(\cdot)$	The convex hull of a vector
$diam(A)$	$\max_{x,y\in A} \\|x-y\\|$
$dist(P,Q)$	$\inf_{x\in P, y\in Q} \\|x-y\\|$
$\|S\|$	The cardinality of a set S
x', x^T	The transpose of a vector x
1	The column vector with all entries equal to 1
$\mathbb{1}_{\{A\}}$	The indicator function of a set A
τ_z^a	The first time that a given random walk hits z starting from a
$H_{\mathcal{Z}}(a,z)$	The expected time that the random walk $\mathcal{Z}$ initiated at a hits z
$H(a,z)$	The expected time that a simple random walk initiated at a hits z
$G_{\tau_z^a}(x)$	The expected number of visits to x before τ_z^a
$\tau(x,y)$	The first time that two given random walks stated from x and y meet each other
$\mathcal{R}(x \leftrightarrow y)$	The effective resistance between two nodes x and y
$c(x,y)$	The conductance of an edge (x,y)
$c(x)$	The sum of all the conductances $c(x,y)$ over neighbors of x
$r(e)$	The resistance of an edge e
$\mathbb{V}(\cdot)$	The voltage function over an electric circuit
$\mathcal{G} = (\mathcal{V}, \mathcal{E})$	An undirected graph with vertex set $\mathcal{V}$ and edge set $\mathcal{E}$
$N(x)$, N_x	The set of neighbors of x
$d(x)$, d_x	The degree of a node x
d_{max}, d_{min}	Maximum and minimum degree of a graph
$d_{\mathcal{G}}(i,j)$	The graphical distance between two nodes i and j
$B_{\mathcal{G}}(i,r)$	A ball of radius r and center i in graph $\mathcal{G}$
$\mathcal{G} \times \mathcal{H}$	Cartesian product of two graphs $\mathcal{G}$ and $\mathcal{H}$

$\mathcal{A}_{\mathcal{G}}$	The adjacency matrix of a graph $\mathcal{G}$
$\mathcal{L}_{\mathcal{G}}$	The Laplacian matrix of a graph $\mathcal{G}$
$\lambda_2(\mathcal{L}_{\mathcal{G}})$	The second smallest eigenvalue of the Laplacian matrix
$\alpha_i(\cdot)$	The ith largest eigenvalue of a symmetric n by n matrix
$\varPhi(S)$	The cut ratio of a set S in a graph $\mathcal{G}$, i.e., $\frac{e(S,S^c)}{\lvert S\rvert\lvert S^c\rvert}$
$\varPhi$	The isoperimetric number of $\mathcal{G}$, i.e., $\min_{S\subset\mathcal{V}}\Phi(S)$
$\perp$	The orthogonality symbol between two vectors
$\mathbb{P}(\cdot)$	Probability of an event
$\mathbb{E}(\cdot)$	Expectation of an event
$\mathcal{M}$	Transition probability matrix of Metropolis chains
λ_{ij}	The Metropolis weight on an edge (i,j), i.e., $\frac{1}{\max\{d(i),d(j)\}}$
$\mathcal{G}(n,p)$	Erdos–Renyi graph of n nodes with probability of emerging an each equal to p
$\sigma_i(P,o)$	The i's nearest node in a network holding object o for a given profile P
$r_i(P)$	The distance between node i and her closest node holding the same resource in P
$\mathcal{N}_k(t)$	The set of all agents at time t whose radius is k
$\mathrm{C}(\mathcal{G})$	The set of proper colorings of a graph $\mathcal{G}$
NE	The set of pure-strategy Nash equilibria of a game
d	The vector of all the degrees of a graph, i.e., $(d_1,\ldots,d_n)$
$\mathcal{B}$	A connected block of a given graph
$\sigma(A)$	The zero pattern of a matrix A
$M_i(t)$	The number of different resources in $B_{\mathcal{G}}(i,r_i)$ at a given time t
R	The effective resistance matrix of a network

List of Figures

Chapter 1
Introduction

Social and distributed networks constitute an important research area which has attracted a lot of attention in the past few years. Such social structures are used to study the relationships between individuals, groups and organizations when they act as members of a society, and have emerged in a variety of fields such as economics, engineering, psychology, sociology and statistics. In particular, with the appearance of game theory and distributed control from one side and myriads of online social networks and availability of huge data sets on the other, modeling different phenomena under social and distributed networks has become an important research topic in recent years.

Traditionally, a common way of modeling different phenomena is to approximate such events with some known linear or static models in order to facilitate the analysis and to utilize a rich collection of results available for such models. Although such simplified models provide ample intuition on understanding the more complex nature of events, in many applications their failure to describe different phenomena has motivated researchers to consider models with nonlinear structures. In fact, the failure of such simplified linear or static models is more pronounced in the study of social networks where the individuals' actions may depend on many factors in the society. However, analyzing such nonlinear models can be much more complicated than analyzing linear ones. Therefore, there is a tradeoff between complexity of a proposed model and the precision of the model to explain a specific phenomenon. This issue has motivated considerable recent research devoted to developing some effective nonlinear models which not only can explain different events with high precision, but also have high tractability capabilities.

Due to the complex nature of social events which might be woven with rational decisions, the existing results related to social and distributed networks are very much case dependent. However, there are still some basic questions that one can ask in this area, and one of them is whether there is any common approach to analyze a broader class of such networks. In other words, one might be interested in some common aspects of the networks which will be preserved as long as the social interactions

S.R. Etesami, *Potential-Based Analysis of Social, Communication, and Distributed Networks*, Springer Theses, DOI 10.1007/978-3-319-54289-8_1

continue. This line of thinking has emerged in the literature under the framework of *potential theory*. In this thesis, we address the above question for several important classes of social and distributed networks, but before we proceed we first review some of the important previous results which effectively model several important phenomena such as diffusion and formation of opinions in social networks, quantized consensus in distributed networks and resource allocation in economic networks.

1.1 Motivation and Past Works

In this section we introduce each of the problems that we consider in the thesis and motivate some of its real world applications. In particular, we provide a comprehensive literature review regarding to each problem.

1.1.1 Quantized Consensus

There has been much recent interest in the design of control protocols for distributed systems, motivated in part by the need to develop protocols for networks of autonomous agents characterized by unavailability of centralized information and time-varying connectivity. A canonical problem in this area is the so-called average consensus problem, wherein a group of agents must agree on the average of their initial values while interacting with neighbors in a (possibly time-varying) network. Protocols for consensus problems must be distributed, relying only on local information at each node, and must be robust to unexpected changes in topology.

It is well understood by now that protocols for consensus play an important role in a number of more sophisticated multi-agent tasks. We mention here distributed optimization [1, 2], coverage control [3], formation control [4, 5], cooperative statistical inference [6, 7], power control [8], load balancing [9], and epidemic routing [10] as examples of control and coordination problems with proposed solutions relying crucially on consensus.

Simplicity and efficiency are two complementary goals in the design of distributed protocols for consensus and other multi-agent problems. On the one hand, one would like to design a simple protocol which is easy to implement and which has little or no reliance on global knowledge about the system. On the other hand, one typically wants to achieve a global goal quickly and efficiently. Interestingly, there often appears to be a tradeoff between convergence speed and locality and simplicity of the algorithm, see [11–14].

Our work is motivated by the realization that often working with real-valued variables in multi-agent control is neither necessary nor efficient. Indeed, limited memory and storage at each agent often forces the variables kept by each agent to lie in a discrete set. We therefore consider the quantized consensus problem, a variation on the consensus problem where the values at the disposal of each agent

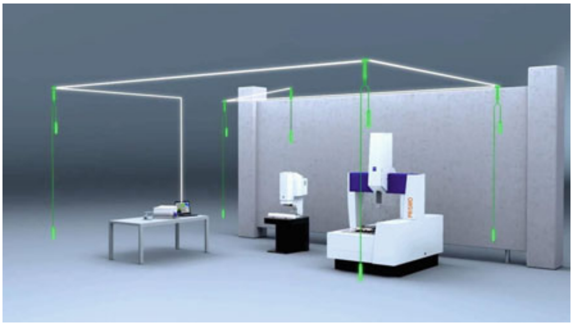


Fig. 1.1 Every sensor measures the room temperature at a different location, and the sensors' goal is to compute the average temperature as quickly as possible

are constrained to be integers lying within a certain range. Previous literature on this problem includes [15–17, 17–23]. As some of the real-world applications of this problem, one can consider information fusion in sensor networks [7], e.g., when every sensor in a sensor network has a different measurement of the temperature and the sensors' goal is to compute the average temperature (Fig. 1.1); multi-agent coordination [21], e.g., when a set of robots coordinate in order to move to the same location; and load balancing in processor networks, which have various applications in computer science [9].

We note that the quantized consensus problem restricts both transmissions to neighbors as well as the values stored by the nodes to be integers. This is related to, but mathematically different from, the related problem of "consensus with finite data rate" wherein nodes store real numbers but transmissions are quantized. We refer the reader to [24–29] for literature on this related problem.

There has been a considerable amount of work on quantized consensus since its introduction in [15], and we make no attempt to provide here a complete survey of the literature. Rather, we summarize the literature which focuses on worst-case convergence times in terms of the number of nodes n in the network.

The original paper [15] contained upper bounds for a natural quantized consensus protocol on a variety of common graphs. Later, the work of [18] proposed a quantized consensus protocol with an upper bound of $O(n^5)$ on the expected convergence time on any fixed graph. For dynamic graphs, [18] obtained a convergence time scaling of $O(n^9)$. The paper [23] obtained an upper bound of $O(n^3)$, but only for complete graphs. A faster upper bound on convergence time and only for static networks was given in the [22], where a protocol was provided whose expected convergence time in general static networks is $O(n^3 \log n)$. As of the writing of this thesis, the upper bounds of $O(n^3 \log n)$ and $O(n^9)$, respectively, are the fastest-known protocols for randomized quantized consensus over static and dynamic networks [18, 22]. It is worth noting that the convergence speed of a protocol is measured by the maximum over all initial inputs of the expected time that the given protocol will run until reaching consensus. Therefore, a faster algorithm will have to have a smaller worst case expected convergence time. We also note that a deterministic version of the quantized-consensus algorithm was introduced in [30], where the authors showed a convergence time of $O(n^2)$ for such dynamics over static networks. However, unlike the randomized quantized consensus protocols considered here as well as in [16, 22, 31], adopting the protocols from [30] to time-varying graphs appears not to be possible.

1.1.2 Hegselmann-Krause Opinion Dynamics

Opinion formation in social networks is an important area of research that has attracted a lot of attention in recent years in a wide range of disciplines, such as psychology, economics, political science, and electrical and computer engineering. A natural question that commonly arises in all those areas is the extent to which one can predict the outcome of the opinion formation of entities under some complex interaction process running among these social actors. As we discussed earlier, the consensus problem is one such attempt to capture the behavioral patterns where the opinions eventually converge to the same value [32–37]. However, there are many situations in which there is neither a desire for consensus nor any tendency for the underlying process to approach a common outcome. In fact, such situations frequently emerge in the context of political elections, product marketing, and capital investment (Fig. 1.2) when there are multiple candidates, product choices, or investment locations among which to select. Those facts have motivated researchers to study disagreement along with consensus.

One of the first studies that considers disagreement beside consensus was undertaken by Friedkin and Johnsen [38], whose model was later extended by Hegselmann and Krause in [39], in the sense that [39] relaxes the assumption of time-invariant influence weights among the agents. More precisely, the Hegselmann-Krause dynamics allow the influence weights to be functions of not only time, but also the states. It is worth noting that although such extensions make the analysis of Hegselmann-Krause dynamics mathematically much more complicated, but interesting, one may argue that the assumption of influence weights depending on the evolving opinion distance (which is the case in the Hegselmann-Krause dynamics) is questionable from a practical point of view, given the literature in experimental social psychology, e.g., see [40, 41], where social psychologists have long been intrigued by the hypothesis that opinion differences reliably predict direct relations of interpersonal influence. Still, a rigorous analysis of the Hegselmann-Krasue dynamics is both theoretically and practically important. The theoretical aspect is that it allows us to develop novel tools useful to study more complex time and state dependent evolutionary dynamics and elaborate on their connections with other fields. The practical aspect is that, other than applications in the modeling of opinion dynamics, the model has applications in the robotics rendezvous problem in plane and space [42]. Accordingly, we consider in this thesis the Hegselmann-Krause model in $\mathbb{R}^d$, where $d \geq 1$.

In the Hegselmann-Krause model, a finite number of agents frequently update their opinions based on the possible interactions among them. The opinion of each agent in this model is captured by a scalar quantity in one dimension or a vector

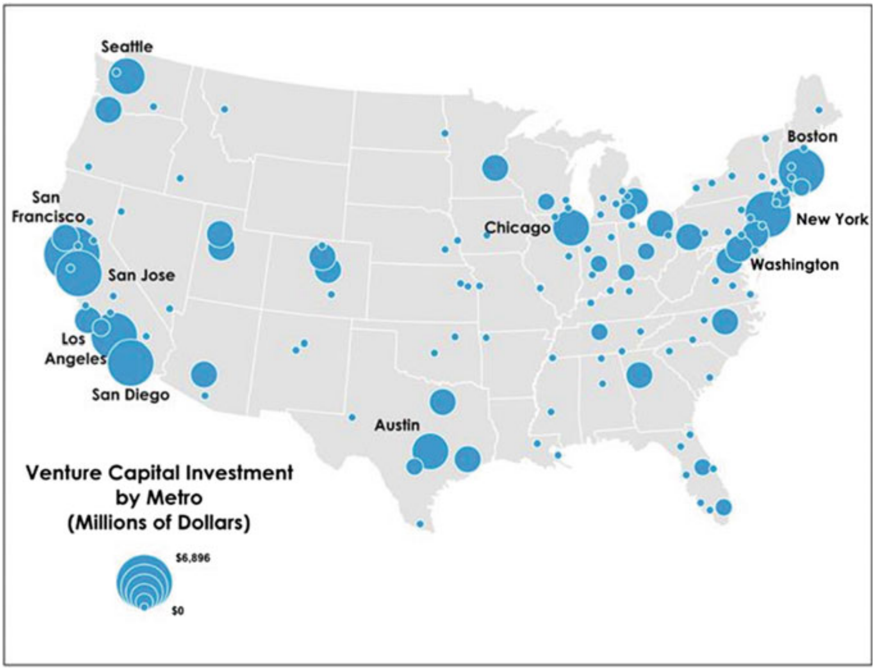


Fig. 1.2 The opinion of where to invest is formed by interaction of capital investors with their friends such that in the long term it determines the concentration of capital in different geographical locations

in Euclidean space $\mathbb{R}^{d>1}$ in higher dimensions. In fact, because of the conservative nature of social entities, each agent in this model communicates only with those whose opinions are closer to him and lie within a certain level of his confidence (bound of confidence), where the distance between agents' opinions is measured by the Euclidian norm in the ambient space. Depending on whether the bound of confidence is the same for all the agents or not, one can distinguish between two different types of dynamics, known as *homogeneous* and *heterogeneous*, respectively. Moreover, the updating process of the agents may be synchronous, meaning that all the agents update simultaneously, or asynchronous, where the agents update in turn. Although at first glance the differences among these four types of dynamics may seem insignificant, in fact, their outcomes are substantially different, such that most of the results from one cannot be carried over to the others [43–45]. In particular, because of the extra freedom for the agents' movements in higher dimensions, analyzing such dynamics for dimensions higher than one is considerably more complex than for the case of one dimension [46–48].

It is known that synchronous homogeneous Hegselmann-Krause dynamics will terminate after finitely many steps [39, 43]. The existing studies on the behavior of the Hegselmann-Krause model in one dimension where the agents' opinions are scalars can be found in [49]. It was shown in [42] that the termination time of the Hegselmann-Krause dynamics in one dimension is at least $O(n)$, where n is the number of agents, and at most $O(n^3)$ [47, 50]. Moreover, the stability and the termination time of such dynamics in higher dimensions were studied in [46, 51], and the work in [46] bounds the termination time of such dynamics using the number of isolated agents through the evolution of the dynamics. In a recent work of Bhattacharyya et al. [47], a polynomial upper bound of $O(n^{10}d^2)$ was given for such dynamics in higher dimensions, but leaving the dependency and improvement of such a bound on the dimension of ambient space as an open problem.

The asynchronous homogeneous Hegselmann-Krause model was considered in [52], where the authors were able to establish stability of this model using a proper quadratic comparison function when the probability of updating for each agent is uniformly bounded from below by some positive constant $p > 0$. In this thesis, we model the evolution of such dynamics as a sequence of best response updates in a potential game and provide a polynomial upper bound for the maximum expected switching topologies and the expected time it takes for the dynamics to reach an arbitrarily small neighborhood of its steady state provided that the agents update uniformly at random. We refer the reader to [53, 54] for some of the possible connections between control of distributed systems and potential games. Furthermore, the synchronous heterogeneous Hegselmann-Krause model was studied in [44], as well as in [45], where the authors conjecture that the number of switching topologies throughout the dynamics must be finite. In fact, our analysis for an asynchronous homogeneous Hegselmann-Krause model here is a step toward more detailed analysis of the heterogeneous model using an appropriate potential function over directed graphs [21, 55]. Furthermore, numerous simulation results have been conducted to study and explore the evolutionary properties of the Hegselmann-Krause dynamics under various settings. For more information, we refer the reader to [39, 43, 44, 56].

1.1.3 Resource Allocation Games

Failures in modeling various events using linear models is more pronounced when there is a sort of competition or rational decision making in the network. This calls for introduction of game theoretic tools, and an investigation of how they can be used in formulating and addressing such problems. In this context, we consider in this thesis a class of problems known as resource allocation problems. In general, distributed network storage games or resource allocation games are characterized by a set of agents who compete for the same set of resources [57, 58], and arise in a wide variety of contexts such as congestion games [59–62], load balancing [9, 63], peer-to-peer systems [64–66], web-caches [67], content management [64], auctioning and mechanism design [68, 69], and market sharing games [70]. Among many problems that arise in such a context, one that stands out is distributed replication, which not only improves the availability of resources for users, but also increases the reliability of the entire network with respect to customer requests [71, 72]. However, one of the main challenges in modeling resource allocation problems using game-theoretic tools is to answer the question of to what extent such models can predict the desired optimal allocations over a given network. Modeling a system as a game, ideally one would like for the set of Nash equilibria of the game to be as close as possible to some target states that one is seeking in the system [73, 74]. In fact, the *price of anarchy (PoA)* [75] is one of the metrics in game theory that measures efficiency and the extent to which a system degrades due to selfish behavior of its agents; it has been widely used in the literature [64, 70, 71, 76].

Distributed replication games with servers that have access to all the resources and are accessible at some cost by users have been studied in [66]. Moreover, the uncapacitated selfish replication game where the agents have access to the set of all resources was studied in [71], where the authors were able to characterize the set of equilibrium points in terms of the parameters of the problem. However, unlike the uncapacitated case, there is no comparable characterization of equilibrium points in capacitated selfish replication (CSR) games. In fact, when the agents have limited capacity, the situation could be much more complicated as the constraint couples the actions of agents much more than in the uncapacitated case or in replication games with servers.

Typically, CSR games are defined in terms of a set of available resources for each player, where the players are allowed to communicate through an undirected communication graph. Such a communication graph identifies the access cost among the players, and the goal for each player is to satisfy his/her customers' needs with minimum cost (Fig. 1.3). Ideally, and in order to avoid any additional cost, each player only wants to use his/her own set of resources. However, due to capacity limitations, the players cannot have immediate access to all the resources they may need, and hence, they will have to borrow some of the resources which are not

Fig. 1.3 Users frequently refer to servers (computers) which are connected through a communication network and request some data. If the servers have the requested data in their cache, they simply provide it to the customers; otherwise, they incur cost by requesting and obtaining data from other servers in the network in order to satisfy their customers' needs

available in their own caches from others in order to meet their customers' demands, and such a transaction incurs cost. In fact, the motivation behind introducing the CSR game is driven by three facts which may arise in most of the resource allocation problems: myopic behavior of the agents aligned with their individual benefit, higher access cost to the resources which are farther away, and budget constraint on the number of resources that each agent can replicate [77, 78]. The problem of finding an equilibrium for CSR games in the case of hierarchical networks was studied in [67]. In this thesis we consider CSR games and develop some simple distributed algorithms to find or approximate their equilibrium points.

1.1.4 Diffusion Game over Social Networks

An important model which is being widely used in order to study competition and rational decision making in social and economic networks is the diffusion model, where the goal is to propagate a certain type of product or behavior in a desired way through a network [79–82]. Other than applications in online advertising for companies' products, such a model has applications in epidemics and immunization vs. virus spreading [83, 84].

In many of the applications in social networks, it is natural to have more than one party that wants to spread information on his own products. This imposes a sort of competition among the providers who are competing for the same set of resources and their goal is to diffuse information on their own product in a desired way across the society [79, 85, 86]. As an example, one can consider a competition between cell-phone companies where each of them is trying to attract the most customers to its own products. Therefore, the question of where such companies can advertise their products and what group of people should they target, who would have more influence in the society, is an important issue. This scenario has been illustrated in Fig. 1.4, where the customers of each of these companies are distinguished by a different color. Such a competition can be modeled within a game theoretic framework, and hence, a natural question one can ask is the characterization of the set of equilibria of such a game. Several papers in the literature have in fact addressed this question in different settings, with some representative ones being [79, 85–88].

Here, we note that due to the complex nature of social events, one can find various models aimed at capturing the idea of competition over social networks. One of the models that describe such a competitive behavior in networks is known as the *competitive diffusion* game [89]. This model can be seen as capturing a competition between two or more parties (types) who initially select a subset of nodes (seeds) to invest, and the goal for each party is to attract as many social entities to his or her own type as possible. It was shown earlier [89] that in general such games do not admit pure-strategy Nash equilibria. It has also been shown in [90] that even on graphs of small diameter such games may not have a pure-strategy Nash equilibrium, but for some special cases they may. In fact, the authors in [90] have shown that for graphs of diameter 2 and under some additional assumptions on the topology of

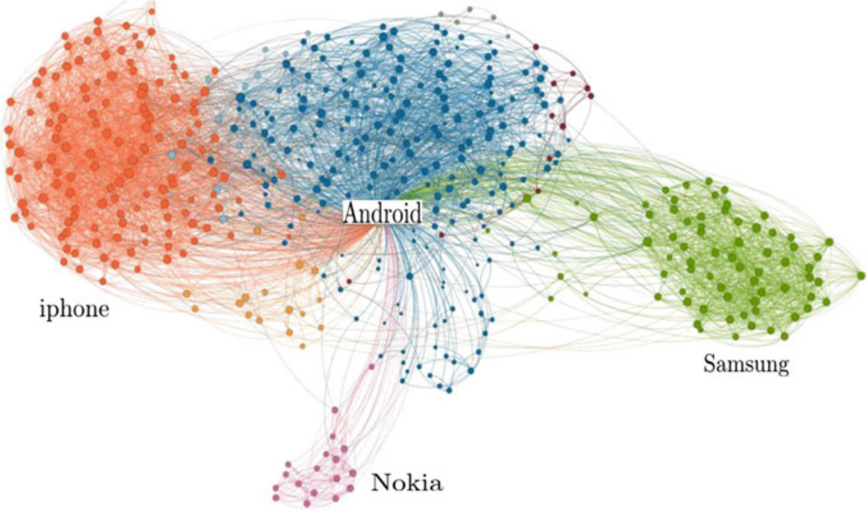


Fig. 1.4 Each cell-phone company is trying to maximize the number of its customers by propagating and advertising its products through the most influencial group of people in the social network. The set of final customers for each type of product is distinguished by a different color

the network, the competitive diffusion game admits a general potential function and hence an equilibrium. However, checking these conditions at the outset for graphs of diameter at most 2 does not seem to be realistic.

In fact, studying the best response of each player is one of the main challenges in general diffusion games. This can be viewed as the best seed placement problem which was extensively studied for different processes [91–94]. However, one of the advantages of the competitive diffusion game model is that it captures the simple fact that being closer to a player's initial seeds will result in adopting that specific player's type. In particular, the adoption rule which is involved in the competitive diffusion game is quite simple such that it enables each player to compute its best response quite fast with respect to others, given that all the other players have fixed their actions. On the other hand, as we will see in this thesis, what makes the analysis of such games more complicated is the behavior of nodes which are equally distanced from the players' seeds. Although there were some recent attempts to characterize these boundary points and show the existence of pure-strategy Nash equilibrium of the diffusion game over different types of networks [89, 95, 96], in the thesis we will address this issue in a more general form, and show that finding an equilibrium for diffusion games is an NP-hard problem over general networks. Therefore, unless P = NP, this strongly suggests that in general the complexity of analyzing such games is a hard task despite its simple adoption rule. It requires additional relaxations in the structure of the game in order to make it more tractable. As one possible approach one may consider a probabilistic version of the diffusion game using some techniques from Markov chains or optimization of harmonic influence centrality [97–99]. We will discuss one such possible relaxation in Appendix B.

1.2 Overview and Contributions

As indicated earlier, this thesis research is devoted to the study of evolutionary behavior of social, communication, and distributed networks, their complexity and convergence rates. Here we provide highlights of the main contributions of the work.

In Chap. 3, we consider an important class of distributed algorithms over networks known as unbiased quantized consensus algorithms, and obtain some strong results on expected time to convergence. In particular, we provide a tight expression for the expected convergence time of unbiased quantized consensus over both general fixed and time-varying networks. We show, using the theory of harmonic functions for reversible Markov chains, that the expected convergence time of such dynamics over static networks lies within a constant factor of the maximum hitting time of an appropriate lazy random walk. This fully characterizes the expected convergence time of unbiased quantized protocol and shows that one cannot obtain a faster convergence rate unless he/she modifies the underlying protocols. We return to this issue again in Chap. 4. Following this, and using the electric resistance analogy of reversible Markov chains, we provide an upper bound of $O(nmD\log n)$ for the expected convergence time to consensus over static networks, where n, m and D, denote, respectively, the number of nodes, the number of edges, and the diameter of the network. Moreover, we show that this upper bound is tight up to a factor of $\log n$ for some simple graphs such as line graph and cycle. Finally, we extend the results to bound the expected convergence time of the underlying dynamics in time-varying networks. Modeling such dynamics as the evolution of a time inhomogeneous Markov chain, we derive an upper bound of $O(n^2 m_{\max} D_{\max} \log^2 n)$ for the expected convergence time of the dynamics using the spectral representation of the networks.

Our main contribution in Chap. 4 with regard to the earlier existing results on quantized consensus problems is to design a protocol wherein nodes cooperate to perform updates on edges connecting them at so-called Metropolis rates. In particular, we are able to improve the existing results on the convergence rate of earlier quantized consensus protocols [18, 22, 31]. More specifically, on fixed graphs, our essentially quadratic convergence time is at least as good as the best earlier known convergence time of [22]. Over dynamic graphs, we improve the best known convergence time given in [31] by a factor of n^2. In fact, our quadratic bounds of $O(n^2 \log n)$ and $O(n^2 \log^2 n)$ given in this chapter for, respectively, static and dynamic networks are better than all previous convergence times for randomized quantized consensus.

In Chap. 5, we consider the Hegselmann-Krause model for opinion dynamics and study the evolution of the system under various settings. We first analyze the termination time of the synchronous Hegselmann-Krause dynamics in arbitrary finite dimensions and show that the termination time in general only depends on the number of agents involved in the dynamics. To the best of our knowledge, this is the sharpest bound for the termination time of such dynamics that connects the convergence speed of the dynamics to the eigenvalues of the adjacency matrix of the connectivity graph in the Hegselmann-Krause dynamics. This answers an open question in [47]: how to obtain a tighter upper bound for the termination time. Furthermore, we study the asynchronous Hegselmann-Krause model from a novel game-theoretic approach and show that the evolution of an asynchronous Hegselmann-Krause model is equivalent to a sequence of better response updates in a well-designed potential game. We then provide a polynomial upper bound for the expected time and expected number of switching topologies until the dynamics reach an arbitrarily small neighborhood of the equilibrium points, provided that the agents update uniformly at random. Finally, we consider the heterogeneous Hegselmann-Krause dynamics and provide a necessary condition for finite termination time of such dynamics.

In Chap. 6, we consider a capacitated selfish replication (CSR) game. Unlike the earlier results on this problem, which in general provide only exponential time search algorithms for finding a pure-strategy Nash equilibrium (NE) unless the number of resources is 2 [67], we show that when the number of resources is less than 5 or the edge density of the network is high enough with respect to the number of resources, there exists an exact polynomial time algorithm which can find a pure-strategy Nash equilibrium. In particular, we consider such games over general undirected networks and devise a quasi-polynomial algorithm which drives the system to an allocation profile whose total cost as well as individual costs lie within a constant factor of that in any pure-strategy NE. We study the price of anarchy of such games over general networks, and show that it is bounded above by 3. We further show that when the underlying network has a tree structure, every globally optimal allocation is an NE, which can be reached in only linear time. We formulate the optimal solutions and NE points of the CSR game using integer linear programs, and provide an upper bound for the minimum social cost of the game using a probabilistic argument. Finally, we introduce the LCSR game as a localized version of the CSR game, wherein the actions of the players are restricted to only their close neighborhoods, and extend all the above results to CSR games with different capacities.

In Chap. 7, we consider the competitive diffusion game, and study the existence of its pure-strategy Nash equilibrium when defined over general undirected networks. We first determine the set of pure-strategy Nash equilibria for two special but well-known classes of networks, namely the lattice and the hypercube. Characterizing the utility of the players in terms of graphical distances of their initial seed placements to other nodes in the network, we show that in general networks the decision process on the existence of pure-strategy NE is an NP-hard problem. Following this, we provide some necessary conditions for a given profile to be an NE. Furthermore, we study players' utilities in the competitive diffusion game over Erdos-Renyi random graphs and show that as the size of the network grows, the utilities of the players are highly concentrated around their expected values, and are bounded below by some threshold based on the parameters of the network. We obtain a lower bound on the maximum social welfare of the game with two players, and study sub-modularity of the players' utilities. Moreover, we provide in Appendix B an electric circuit formulation of the diffusion game on general networks. This model allows us to analyze the diffusion game in more detail and within a more practical setting.

References

1. J. N. Tsitsiklis, D. P. Bertsekas, and M. Athans, "Distributed asynchronous deterministic and stochastic gradient optimization algorithms," *IEEE Transactions on Automatic Control*, vol. 31, no. 9, pp. 803–812, 1986.
2. A. Nedic and A. Ozdaglar, "Distributed subgradient methods in multi-agent optimization," *IEEE Transactions on Automatic Control*, vol. 54, no. 1, pp. 48–61, 2009.
3. C. Gao, J. Cortes, and F. Bullo, "Notes on averaging over acyclic digraphs and discrete coverage control," *Automatica*, vol. 44, no. 8, pp. 2120–2127, 2008.
4. W. Ren and R. W. Beard, *Distributed Consensus in Multi-Vehicle Cooperative Control*. London: Springer, 2008.
5. A. Olshevsky, "Efficient information aggregation strategies for distributed control and signal processing," *Ph. D. thesis*, arXiv:1009.6036, 2010.
6. M. Rabbat and R. D. Nowak, "Distributed optimization in sensor networks," in *Information Processing and Sensor Networks*. IEEE, 2004, pp. 29–27.
7. L. Xiao, S. Boyd, and S.-J. Kim, "Distributed average consensus with least-mean-square deviation," *Journal of Parallel and Distributed Computing*, vol. 67, no. 1, pp. 33–46, 2007.
8. S. Ram, V. Veeravalli, and A. Nedic, "Distributed non-autonomous power control through distributed convex optimization," in *INFOCOM*. IEEE, 2008, pp. 3001–3005.
9. B. Ghosh and S. Muthukrishnan, "Dynamic load balancing in parallel and distributed networks by random matchings," in *Proceedings of the Sixth Annual ACM Symposium on Parallel Algorithms and Architectures*. ACM, 1994, pp. 226–235.
10. G. Neglia, G. Reina, and S. Alouf, "Distributed gradient optimization for epidemic routing: a preliminar evaluation," in *Wireless Days, 2nd IFIP*. IEEE, 2009, pp. 1–6.
11. D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation*. Old Tappan, NJ (USA); Prentice Hall Inc, 1989.
12. L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Systems and Control Letters*, p. 6517, 2004.
13. S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Gossip algorithms: Design, analysis and applications," in *Proceedings IEEE INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies*, vol. 3. IEEE, 2005, pp. 1653–1664.
14. M. Draief and M. Vojnovic, "Convergence speed of binary interval consensus," *SIAM Journal on Control and Optimization*, vol. 50, no. 3, pp. 1087–1109, 2012.
15. A. Kashyap, T. Başar, and R. Srikant, "Quantized consensus," *Automatica*, vol. 43, no. 7, pp. 1192–1203, 2007.
16. S. Etesami and T. Başar, "Convergence time for unbiased quantized consensus," *Proc. 52nd IEEE Conference on Decision and Control. CDC'13, Dec 10-13, 2013; Florence, Italy*, pp. 6190–6195, 2013.
17. R. Carli, F. Fagnani, P. Frasca, and S. Zampieri, "Gossip consensus algorithms via quantized communication," *Automatica*, vol. 46, no. 1, pp. 70–80, 2010.
18. M. Zhu and S. Martinez, "On the convergence time of asynchronous distributed quantized averaging algorithms," *IEEE Transactions on Automatic Control*, vol. 56, no. 2, pp. 386–390, 2011.
19. K. Cai and H. Ishii, "Quantized consensus and averaging on gossip digraphs," *IEEE Transactions on Automatic Control*, vol. 56, no. 9, pp. 2087–2100, 2011.
20. F. Bénézit, P. Thiran, and M. Vetterli, "Interval consensus: from quantized gossip to voting," in *IEEE International Conference on Acoustics, Speech and Signal Processing, 2009. ICASSP 2009*. IEEE, 2009, pp. 3661–3664.
21. R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1520–1533, 2004.
22. S. Shang, P. Cuff, P. Hui, and S. Kulkarni, "An upper bound on the convergence time for quantized consensus," in *2013 Proceedings INFOCOM*. IEEE, 2013, pp. 600–604.

23. K. Cai and H. Ishii, “Convergence time analysis of quantized gossip consensus on digraphs,” *Automatica*, vol. 48, pp. 2344–2351, 2012.
24. R. Carli, F. Bullo, and S. Zampieri, “Quantized average consensus via dynamic coding/decoding schemes,” *International Journal of Robust and Nonlinear Control*, vol. 20, no. 2, pp. 156–175, 2010.
25. T. Li and L. Xie, “Distributed coordination of multi-agent systems with quantized-observer based encoding-decoding,” *IEEE Transactions on Automatic Control*, vol. 57, no. 12, pp. 3023–3027, 2012.
26. M. Guo and D. Dimarogonas, “Consensus with quantized relative state measurements,” *Automatica*, vol. 49, no. 8, pp. 2531–2537, 2013.
27. J. Lavaei and R. M. Murray, “Quantized consensus by means of gossip algorithm,” *IEEE Transactions on Automatic Control*, vol. 57, no. 1, pp. 19–32, 2012.
28. A. Olshevsky, “Consensus with ternary messages,” *SIAM Journal on Control and Optimization*, vol. 52, no. 2, pp. 987–1009, 2014.
29. M. El Chamie, J. Liu, and T. Başar, “Design and analysis of distributed averaging with quantized communication,” *Proc. 53rd IEEE Conference on Decision and Control (CDC’14, Dec 15-17, 2014; Los Angeles, CA)*, pp. 3860–3865, 2014.
30. J. M. Hendrickx, A. Olshevsky, and J. N. Tsitsiklis, “Distributed anonymous discrete function computation,” *IEEE Transactions on Automatic Control*, vol. 56, no. 10, pp. 2276–2289, 2011.
31. S. R. Etesami and T. Başar, “Convergence time for unbiased quantized consensus over static and dynamic networks,” *IEEE Transactions on Automatic Control, 61(2), pp. 443-455, February*, 2016.
32. T. C. Aysal, A. D. Sarwate, and A. G. Dimakis, “Reaching consensus in wireless networks with probabilistic broadcast,” *Proceedings of the 47th Annual Allerton Conference on Communication, Control, and Computing*, pp. 732–739, 2009.
33. S. Sundaran and C. N. Hadjicostis, “Finite-time distributed consensus in graphs with time-invariant topologies,” *Proc. American Control Conference*, pp. 711–716, 2007.
34. M. H. DeGroot, “Reaching a consensus,” *Journal of the American Statistical Association*, vol. 69, no. 345, pp. 118–121, 1974.
35. A. Olshevsky and J. Tsitsiklis, “On the nonexistence of quadratic Lyapunov functions for consensus algorithms,” *IEEE Transactions on Automatic Control*, vol. 53, pp. 2642–2645, 2008.
36. J. C. Delvenne, R. Carli, and S. Zampieri, “Optimal strategies in the average consensus problem,” *Wavelet Analysis and Multiresolution Methods*, vol. 56, pp. 759–765, 2009.
37. L.Wang and F. Xiao, “Finite-time consensus problems for networks of dynamic agents,” *IEEE Transaction On Automatic Control*, vol. 55, no. 5, 2010.
38. N. E. Friedkin and E. C. Johnsen, “Social influence networks and opinion change,” *Advances in Group Processes*, vol. 16, no. 1, pp. 1–29, 1999.
39. R. Hegselmann and U. Krause, “Opinion dynamics and bounded confidence models, analysis, and simulation,” *Artificial Societies and Social Simulation*, vol. 5, pp. 1–33, 2002.
40. J. B. Stiff and P. A. Mongeau, *Persuasive Communication*. Guilford Press, 2003.
41. N. E. Friedkin and E. C. Johnsen, *Social Influence Network Theory*. Cambridge University Press, New York, 2011.
42. F. Bullo, J. Cortes, and S. Martinez, *Distributed Control of Robotic Networks*. Princeton University Press, 2009.
43. J. Lorenz, *Repeated averaging and bounded-confidence, modeling, analysis and simulation of continuous opinion dynamics*. Ph.D. Dissertation, University of Bremen, 2007.
44. J. Lorenz, “Heterogeneous bounds of confidence: Meet, discuss and find consensus!” *Complexity*, vol. 15, no. 4, pp. 43–52, 2010.
45. A. Mirtabatabaei and F. Bullo, “Opinion dynamics in heterogeneous networks: Convergence conjectures and theorems,” *SIAM Journal on Control and Optimization*, vol. 50, no. 5, pp. 2763–2785, 2012.
46. S. R. Etesami, T. Başar, A. Nedić, and B. Touri, “Termination time of multidimensional Hegselmann-Krause opinion dynamics,” in *Proc. American Control Conference (ACC)*. IEEE, 2013, pp. 1255–1260.

47. A. Bhattacharyya, M. Braverman, B. Chazelle, and H. L. Nguyen, “On the convergence of the Hegselmann-Krause system,” in *Proceedings of the 4th Conference on Innovations in Theoretical Computer Science*. ACM, 2013, pp. 61–66.
48. B. Chazelle, “The total s-energy of a multiagent system,” *SIAM Journal on Control and Optimization*, vol. 49, no. 4, pp. 1680–1706, 2011.
49. S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, “Gossip algorithms: Design, analysis, and applications,” *IEEE INFOCOM*, vol. 3, pp. 1653–1664, 2005.
50. S. Mohajer and B. Touri, “On convergence rate of scalar Hegselmann-Krause dynamics,” in *Proc. American Control Conference (ACC)*. IEEE, 2013, pp. 206–210.
51. A. Nedić and B. Touri, “Multi-dimensional Hegselmann-Krause dynamics,” in *Proc. 2012 IEEE 51st Annual Conference on Decision and Control (CDC)*. IEEE, 2012, pp. 68–73.
52. B. Touri and C. Langbort, “On endogenous random consensus and averaging dynamics,” *Proc. 52nd IEEE Conference on Decision and Control (CDC)*, pp. 6208–6212, 2013.
53. J. R. Marden, G. Arslan, and J. S. Shamma, “Cooperative control and potential games,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 39, no. 6, pp. 1393–1407, 2009.
54. A. Rantzer, “Using game theory for distributed control engineering,” *Language*, vol. 280, no. 53, p. 16, 2008.
55. H. Zhang, F. L. Lewis, and Z. Qu, “Lyapunov, adaptive, and optimal design techniques for cooperative systems on directed communication graphs,” *IEEE Transactions on Industrial Electronics*, vol. 59, no. 7, pp. 3026–3041, 2012.
56. J. M. Hendrickx, *Graphs and networks for the analysis of autonomous agent systems*. Ph.D. dissertation, Université Catholique de Louvain, 2011.
57. V. Pacifici and G. Dan, “Convergence in player-specific graphical resource allocation games,” *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 11, pp. 2190–2199, 2012.
58. A. M. Masucci and A. Silva, “Strategic resource allocation for competitive influence in social networks,” *arXiv preprint* arXiv:1402.5388, 2014.
59. I. Milchtaich, “Congestion games with player-specific payoff functions,” *Games and Economic Behavior*, vol. 13, no. 1, pp. 111–124, 1996.
60. H. Ackermann, H. Röglin, and B. Vöcking, “On the impact of combinatorial structure on congestion games,” *Journal of the ACM (JACM)*, vol. 55, no. 6, p. 25, 2008.
61. A. Fabrikant, C. Papadimitriou, and K. Talwar, “The complexity of pure Nash equilibria,” in *Proceedings of the Thirty-Sixth Annual ACM Symposium on Theory of Computing*. ACM, 2004, pp. 604–612.
62. V. Anantharam, “On the Nash dynamics of congestion games with player-specific utility,” in *Proc. 43rd IEEE Conference on Decision and Control, 2004. CDC*, vol. 5. IEEE, 2004, pp. 4673–4678.
63. I. Caragiannis, M. Flammini, C. Kaklamanis, P. Kanellopoulos, and L. Moscardelli, “Tight bounds for selfish and greedy load balancing,” in *Automata, Languages and Programming*. Springer, 2006, pp. 311–322.
64. G. G. Pollatos, O. A. Telelis, and V. Zissimopoulos, “On the social cost of distributed selfish content replication,” in *Ad Hoc and Sensor Networks, Wireless Networks, Next Generation Internet*. Springer, 2008, pp. 195–206.
65. E. J. Friedman, J. Y. Halpern, and I. Kash, “Efficiency and Nash equilibria in a scrip system for P2P networks,” in *Proceedings of the 7th ACM Conference on Electronic Commerce*. ACM, 2006, pp. 140–149.
66. N. Laoutaris, O. Telelis, V. Zissimopoulos, and I. Stavrakakis, “Distributed selfish replication,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 17, no. 12, pp. 1401–1413, 2006.
67. R. Gopalakrishnan, D. Kanoulas, N. N. Karuturi, C. P. Rangan, R. Rajaraman, and R. Sundaram, “Cache me if you can: capacitated selfish replication games,” in *LATIN 2012: Theoretical Informatics*. Springer, 2012, pp. 420–432.
68. R. T. Maheswaran and T. Başar, “Nash equilibrium and decentralized negotiation in auctioning divisible resources,” *Group Decision and Negotiation*, vol. 12, no. 5, pp. 361–395, 2003.

69. R. T. Maheswaran and T. Başar, “Decentralized network resource allocation as a repeated noncooperative market game,” in *Proceedings of the 40th IEEE Conference on Decision and Control*, vol. 5. IEEE, 2001, pp. 4565–4570.
70. M. X. Goemans, L. Li, V. S. Mirrokni, and M. Thottan, “Market sharing games applied to content distribution in ad hoc networks,” *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 5, pp. 1020–1033, 2006.
71. B.-G. Chun, K. Chaudhuri, H. Wee, M. Barreno, C. H. Papadimitriou, and J. Kubiatowicz, “Selfish caching in distributed systems: a game-theoretic analysis,” in *Proceedings of the Twenty-Third Annual ACM Symposium on Principles of Distributed Computing*. ACM, 2004, pp. 21–30.
72. S. Goyal and F. Vega-Redondo, “Learning, network formation and coordination,” Tinbergen Institute, Tech. Rep., 2000.
73. R. Johari, S. Mannor, and J. N. Tsitsiklis, “Efficiency loss in a network resource allocation game: the case of elastic supply,” *IEEE Transactions on Automatic Control*, vol. 50, no. 11, pp. 1712–1724, 2005.
74. J. R. Marden and T. Roughgarden, “Generalized efficiency bounds in distributed resource allocation,” in *Proc. 49th IEEE Conference on Decision and Control (CDC)*. IEEE, 2010, pp. 2233–2238.
75. E. Koutsoupias and C. Papadimitriou, “Worst-case equilibria,” in *STACS 99*. Springer, 1999, pp. 404–413.
76. A. Vetta, “Nash equilibria in competitive societies, with applications to facility location, traffic routing and auctions,” in *Proceedings. 43rd Annual IEEE Symposium on Foundations of Computer Science*. IEEE, 2002, pp. 416–425.
77. J. R. Correa, A. S. Schulz, and N. E. Stier-Moses, “Selfish routing in capacitated networks,” *Mathematics of Operations Research*, vol. 29, no. 4, pp. 961–976, 2004.
78. N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani, *Algorithmic Game Theory*. Cambridge University Press Cambridge, 2007, vol. 1.
79. S. Goyal and M. Kearns, “Competitive contagion in networks,” in *Proceedings of the 44th Symposium on Theory of Computing*. ACM, 2012, pp. 759–774.
80. H. P. Young, “The diffusion of innovations in social networks,” *Economy as an Evolving Complex System. Proceedings Volume in the Santa Fe Institute Studies in the Sciences of Complexity*, vol. 3, pp. 267–282, 2002.
81. 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*. ACM, 2003, pp. 137–146.
82. D. Acemoglu, A. Ozdaglar, and E. Yildiz, “Diffusion of innovations in social networks,” in *Proc. 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*. IEEE, 2011, pp. 2329–2334.
83. A. Khanafer, T. Başar, and B. Gharesifard, “Stability properties of infected networks with low curing rates,” in *Proc. 2014 American Control Conference (ACC 2014), Portland, OR*. IEEE, 2014, pp. 3591–3596.
84. A. Khanafer and T. Başar, “Information spread in networks: control, games, and equilibria,” in *Proc. Information Theory and Applications Workshop (ITA’14), San Diego, California*, 2014.
85. S. Bharathi, D. Kempe, and M. Salek, “Competitive influence maximization in social networks,” in *Internet and Network Economics*. Springer, 2007, pp. 306–311.
86. S. Brânzei and K. Larson, “Social distance games,” in *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence-Volume One*. AAAI Press, 2011, pp. 91–96.
87. M. Richardson and P. Domingos, “Mining knowledge-sharing sites for viral marketing,” in *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2002, pp. 61–70.
88. Y. Singer, “How to win friends and influence people, truthfully: influence maximization mechanisms for social networks,” in *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining*. ACM, 2012, pp. 733–742.

89. N. Alon, M. Feldman, A. D. Procaccia, and M. Tennenholtz, “A note on competitive diffusion through social networks,” *Information Processing Letters*, vol. 110, no. 6, pp. 221–225, 2010.
90. R. Takehara, M. Hachimori, and M. Shigeno, “A comment on pure-strategy Nash equilibria in competitive diffusion games,” *Information Processing Letters*, vol. 112, no. 3, pp. 59–60, 2012.
91. E. Yildiz, D. Acemoglu, A. Ozdaglar, A. Saberi, and A. Scaglione, “Discrete opinion dynamics with stubborn agents,” *Available at SSRN 1744113*, 2011.
92. M. Fazli, M. Ghodsi, J. Habibi, P. J. Khalilabadi, V. Mirrokni, and S. S. Sadeghabad, “On the non-progressive spread of influence through social networks,” in *LATIN 2012: Theoretical Informatics*. Springer, 2012, pp. 315–326.
93. E. Ackerman, O. Ben-Zwi, and G. Wolfovitz, “Combinatorial model and bounds for target set selection,” *Theoretical Computer Science*, vol. 411, no. 44, pp. 4017–4022, 2010.
94. A. Gionis, E. Terzi, and P. Tsaparas, “Opinion maximization in social networks.” in *SDM*. SIAM, 2013, pp. 387–395.
95. L. Small and O. Mason, “Nash equilibria for competitive information diffusion on trees,” *Information Processing Letters*, vol. 113, no. 7, pp. 217–219, 2013.
96. L. Small and O. Mason, “Information diffusion on the iterated local transitivity model of online social networks,” *Discrete Applied Mathematics*, 2012.
97. D. Acemoglu, G. Como, F. Fagnani, and A. Ozdaglar, “Opinion fluctuations and disagreement in social networks,” *Mathematics of Operations Research*, vol. 38, no. 1, pp. 1–27, 2013.
98. D. Acemoglu, G. Como, F. Fagnani, and A. Ozdaglar, “Message passing optimization of harmonic influence centrality,” *ACM SIGMETRICS Performance Evaluation Review, vol. 42(3)*, 2014.
99. L. Vassio, F. Fagnani, P. Frasca, and A. Ozdaglar, “Harmonic influence in large-scale networks,” *IEEE Transactions on Control of Network Systems, vol. 1(1)*, 2014.

Chapter 2
Notation and Mathematical Preliminaries

In this chapter, we first introduce the notations that will be used throughout the thesis. After that, we discuss and state some known results, as background material relevant to our future development.

2.1 Notation

The notations used in this thesis are picked to be as intuitive as possible. Readers may skip this section in a first reading, and then come back to it as needed.

2.1.1 Sets, Vectors and Matrices

We let $\mathbb{N}$, $\mathbb{Z}$, and $\mathbb{R}$ denote, respectively, the sets of positive integers, integers, and real numbers. Given an integer $k > 0$, we denote the set of all k-tuples of integers by $\mathbb{Z}^k$. For a vector $v \in \mathbb{R}^n$, we let v_i be the ith entry of v, and v^T (occasionally v') be the transpose of v. Similarly, for a matrix A, we let $A(i, j)$ (occasionally A_{ij}) be the ijth entry of A. We use $\|v\|$ to denote the Euclidean norm of a vector v. For any two vectors $u, v \in \mathbb{Z}^k$, we let $u \oplus v$ be their sum vector in mod 2, i.e., $(u \oplus v)_i = (u_i + v_i) \mod 2$, for all $i = 1, \dots, k$. We let e_i denote the standard ith unit Euclidean vector in $\mathbb{R}^n$ (with 1 in ith location and 0's elsewhere). For a positive integer n, we let $[n] := \{1, \dots, n\}$. We let $\mathbf{1}$ be a column vector with all entries equal to 1. We say that a vector v is *stochastic* if $v_i \geq 0$ for all $i \in [n]$ and $\sum_{i=1}^{n} v_i = 1$. Similarly, we say that a matrix A is stochastic if each row of A is a stochastic vector, and we let $\min^+ A = \min_{i,j}\{A_{ij} | A_{ij} > 0\}$. If both A and A^T are stochastic, we say that A is doubly stochastic. A *scrambling matrix* is a stochastic matrix with the additional property that the inner product of each pair of its rows is

S.R. Etesami, *Potential-Based Analysis of Social, Communication, and Distributed Networks*, Springer Theses, DOI 10.1007/978-3-319-54289-8_2

positive. For a vector y we use $conv(y)$ to denote the convex hull of its components and $diam(conv(y)) = \max_{p,q \in conv(y)} \|p - q\|$. We define the distance between two sets $P, Q \subseteq \mathbb{R}^n$ to be $dist(P, Q) = \inf_{p \in P, q \in Q} \|p - q\|$. For a vector $(s_1, s_2, \dots, s_n)$, we sometimes write (s_i, s_{-i}), where s_{-i} is the set of all entries except the ith one. For a real number a we let $\lfloor a \rfloor$ be the largest integer less than or equal to a. Similarly, we denote the smallest integer greater than or equal to a by $\lceil a \rceil$. Finally, we use $|S|$ to denote the cardinality of a finite set S.

2.1.2 Markov Chains and Electric Circuits

A Markov chain is a stochastic process that satisfies the Markov property, i.e., conditioned on the present state of the system, its future and past are independent. If the state space is finite, the transition probability distribution of the chain can be represented by a matrix P, called the transition matrix, with P_{ij} being the probability that the chain will move to state j, given that it is currently at state i. A Markov chain with transition matrix P is *reversible* if there is a probability distribution $\pi(\cdot)$ over its states such that $\pi_i P_{ij} = \pi_j P_{ji}$ for all states i and j. A Markov chain is said to be *irreducible* if there is a positive probability of reaching any state from any other state. A random walk on a graph $\mathcal{G}$ with transition probability matrix P is a Markov chain with transition matrix P over the state space of all the vertices. For a random walk $\mathcal{Z}$ with transition probability matrix P, we let the *hitting time* random variable τ_y^x denote the first time that the random walk initiated from x hits the state y. Also, we let $H_{\mathcal{Z}}(x, y)$ denote the expected hitting time that the random walk $\mathcal{Z}$ initiated at x hits y for the first time, i.e., $H_{\mathcal{Z}}(x, y) = \mathbb{E}[\tau_y^x]$. We take $G_{\tau_y^x}(z)$ to be the expected number of visits to z before τ_y^x when the random walk has been initiated from x. Similarly, for two random walks moving based on some joint distribution, we let the meeting time random variable $\tau(x, y)$ be the first time until the random walks meet, and we refer to its expected value, denoted by $M(x, y) = \mathbb{E}[\tau(x, y)]$ as the expected meeting time, given that the chains have been originated from x and y. We denote by $\mathbb{V}(\cdot)$ the voltage function over an electric circuit for two distinguished nodes a and z as terminal nodes such that $\mathbb{V}(a) = 0$ and $\mathbb{V}(z) = 1$, and let $\mathbb{V}_{xy} = \mathbb{V}(x) - \mathbb{V}(y)$. We let $\mathcal{R}(x \leftrightarrow y)$ be the effective resistance between two nodes x and y in an electric network when every edge has resistance equal to 1, i.e., the induced voltage between terminals x and y when a unit current is inserted into x and is evicted from y.

2.1.3 Graph Theory

We use $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ for an undirected network (graph) with a node set $\mathcal{V}$ and an edge set $\mathcal{E}$. For an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, we let $N(x)$ (occasionally N_x) be the set of neighbors of x and $d(x) = |N(x)|$ (occasionally d_x) be the degree of node x. For any two nodes $i, j \in \mathcal{V}$, we let $d_{\mathcal{G}}(i, j)$ be the graphical distance between them, that is, the length of the shortest path which connects i and j. The diameter of the graph, denoted by D, is the maximum distance between any pair of vertices, that is, $D = \max_{u,v \in \mathcal{V}(\mathcal{G})} d_{\mathcal{G}}(u, v)$. Moreover, for an arbitrary node $i \in \mathcal{V}$ and an integer $r \geq 0$, we define a graphical ball of radius r and center i to be the set of all the nodes in the graph $\mathcal{G}$ whose graphical distance is at most r to the node i, i.e., $B_{\mathcal{G}}(i, r) = \{x \in \mathcal{V} | d_{\mathcal{G}}(i, x) \leq r\}$. Also, for a set of vertices $S \subseteq \mathcal{V}$ and a vertex x, we let $d_{\mathcal{G}}(x, S) = \min_{y \in S}\{d_{\mathcal{G}}(x, y)\}$. We let $\mathcal{G} \times \mathcal{G} = (\mathcal{V} \times \mathcal{V}, \mathcal{E}')$ be the Cartesian product of $\mathcal{G}$ with itself, i.e., $\big\{(x, y), (r, s)\big\} \in \mathcal{E}'$ if and only if $x = r, s \in N(y)$ or $y = s, r \in N(x)$. For a graph $\mathcal{G}$, we let $\mathcal{A}_{\mathcal{G}}$ be its adjacency matrix, i.e., $\mathcal{A}_{\mathcal{G}}(i, j) = 1$ if and only if $(i, j) \in \mathcal{E}$ and $\mathcal{A}_{\mathcal{G}}(i, j) = 0$, otherwise. We also let $\mathcal{D}_{\mathcal{G}}$ be a diagonal matrix whose diagonal entries are equal to the degree of the nodes in the graph. Moreover, we denote the Laplacian of this graph by $\mathcal{L}_{\mathcal{G}} = \mathcal{D}_{\mathcal{G}} - \mathcal{A}_{\mathcal{G}}$.

2.2 Relevant Results on Markov Chains

In this section, we discuss some relevant useful results from the theory of Markov chains which will be used to prove some of our main results in the future chapters. We start our discussion by defining some additional notation that will be used throughout the thesis.

A simple random walk on a graph $\mathcal{G}$ is a Markov chain over the state space of all the vertices $\mathcal{V}$ with transition probabilities

$$P(x, y) = \begin{cases} \frac{1}{d(x)}, & \text{if } \ y \in N(x) \\ 0, & \text{otherwise}, \end{cases}$$

where $d(x)$ denotes the degree of a node x in the graph $\mathcal{G}$. Note that a simple random walk is a special case of a weighted random walk when the weights of all edges in $\mathcal{G}$ are equal to 1. It is well known that every reversible Markov chain is a weighted random walk on a network. Suppose P is a transition matrix of a Markov chain on a finite set S, which is reversible with respect to a probability distribution $\pi(\cdot)$. Define conductance on edges by $c(x, y) = \pi(x)P(x, y)$ and introduce $c(x) := \sum_{y:\ y \in N(x)} c(x, y)$. Also, the resistance of each edge e is defined to be the inverse of conductance, i.e., $r(e) = \frac{1}{c(e)}$.

Lemma 2.1 *$\frac{G_{\tau_z^a}(x)}{d(x)}$ is equal to the induced voltage between x and z, i.e., $\mathbb{V}_{xz}$ when we define the terminal voltages to be $\mathbb{V}_{zz} = 0$, $\mathbb{V}_{az} = \frac{G_{\tau_z^a}(a)}{d(a)}$. Moreover, for all x we have*

$$\frac{1}{2}\big[\mathcal{R}(a \leftrightarrow z) + \mathcal{R}(z \leftrightarrow x) - \mathcal{R}(a \leftrightarrow x)\big] = \frac{G_{\tau_z^a}(x)}{d(x)} = \mathbb{V}_{xz}.$$

Proof This is the result of Corollary 3 in [1]. □

By taking summation over the above equality and noting that $\sum_x G_{\tau_z^a}(x)$ is equal to the expected hitting time of a simple random walk when it starts from a and hits z, we get:

$$H(a,z) = \frac{1}{2}\sum_x d(x)\big[\mathcal{R}(a \leftrightarrow z) + \mathcal{R}(z \leftrightarrow x) - \mathcal{R}(a \leftrightarrow x)\big], \qquad (2.1)$$

where $H(\cdot,\cdot)$ is the expected hitting time function of the simple random walk.

Lemma 2.2 (Random Target Lemma) *For an irreducible Markov chain with state space $\Omega = \{1, 2, \ldots, n\}$, transition matrix P, and stationary distribution π (i.e., a stochastic vector such that $\pi' P = \pi'$), we have*

$$\sum_{j=1}^{n} \pi_j H_{\mathrm{P}}(i,j) = \sum_{k=2}^{n} \frac{1}{1-\alpha_k(P)}, \quad \forall i \in \Omega$$

where $1 = \alpha_1(P) > \alpha_2(P) \geq \alpha_3(P) \geq \ldots, \geq \alpha_n(P)$ denote the eigenvalues of P in a non-increasing order and $H_{\mathrm{P}}(\cdot,\cdot)$ is the expected hitting time function of the chain.

Proof The idea of the proof is to find a recursion matrix equality for the expected hitting time of a chain with transition probability matrix P and interpreting the solution of this equation based on the eigenvalues of P. A complete proof can be found in [2] and also [3]. □

Lemma 2.3 ([4, Chap. 2]) *Let $\tau(x,y)$ denote the first time until two continuous-time random walks moving based on some joint distribution meet each other, given that the walks started from x and y. Then we have,*

$$\mathbb{P}\big(\tau(x,y) > t\big) \leq e^{-\lfloor \frac{t}{eM} \rfloor},$$

where $M = \max_{r,s} \mathbb{E}[\tau(r,s)]$ is the maximum expected meeting time of the process.

Definition 2.1 A function $h : \Omega \to \mathbb{R}$ is called harmonic at a vertex $x \in \Omega$ for a Markov chain with transition probability matrix P, if $h(x) = \sum_{y\in\Omega} P(x,y)h(y)$.

Lemma 2.4 *Let $\{X_t\}$ be a Markov chain with an irreducible transition matrix P, let $B \subset \Omega$, and $h_B : B \to R$ be a function defined on B. The function $h : \Omega \to R$ defined by $h(x) := \mathbb{E}[h_B(X_{\tau_B^x})]$ is the unique extension of h_B such that $h(x) = h_B(x)$ for all $x \in B$ and h is harmonic for P at all $x \in \Omega \backslash B$.*

Proof The proof can be found in [5]. □

Remark 1 Given a nonempty subset $B \subset \Omega$ and a Markov chain with an irreducible transition matrix P, every harmonic function $h(\cdot) : \Omega \to \mathbb{R}$ over $\Omega \backslash B$ which satisfies $h(x) = 0, \forall x \in B$ (respectively, $h(x) \geq 0, \forall x \in B$), must be identical to zero (respectively, must be nonnegative) over the entire Ω [5].

2.3 Relevant Results on Spectral Graph Theory

In this section, we briefly discuss some relevant useful results from spectral graph theory.

Lemma 2.5 ([6, Perron-Frobenius for Laplacians]) *Let $\mathcal{L}$ be a matrix with non-positive off-diagonal entries such that the graph of the non-zero off-diagonal entries is connected. Then the smallest eigenvalue has multiplicity 1, and the corresponding eigenvector is strictly positive.*

Next, we state Cheeger's inequality, which relates the spectral gap of the Laplacian matrix to the expansion of its corresponding graph.

Lemma 2.6 ([7, Cheeger's Inequality]) *Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be an undirected graph with Laplacian matrix $\mathcal{L}$. Moreover, for a subset of vertices $S \subseteq \mathcal{V}$, let $e(S, S^c)$ denote the number of edges with one vertex in S and one vertex in its complement S^c. Defining the cut ratio $\Phi(S) = \frac{e(S,S^c)}{|S||S^c|}$ and isoperimetric number of $\mathcal{G}$ by $\Phi = \min_{S\subset\mathcal{V}} \Phi(S)$, we have*

$$\frac{\Phi^2}{2d_{max}} \leq \lambda_2(\mathcal{L}) \leq 2\Phi,$$

where d_{max} denotes the maximum degree of the graph $\mathcal{G}$ and $\lambda_2(\mathcal{L})$ is the second smallest eigenvalue of the Laplacian $\mathcal{L}$.

Lemma 2.7 ([8, Courant-Fischer Formula]) *Let A be an $n \times n$ symmetric matrix with eigenvalues $\lambda_1 \leq \lambda_2 \leq \ldots, \leq \lambda_n$ and corresponding eigenvectors $v_1, \ldots, v_n$. Moreover, for $1 \leq k \leq n$, let S_k denote the span of $v_1, \ldots, v_k$ (with $S_0 = \{0\}$), and let $S_k^\perp$ denote the orthogonal complement of S_k, i.e., $S_k^\perp = \{v \in \mathbb{R}^n | v'u = 0, \forall u \in S_k\}$. Then*

$$\lambda_k = \min_{\substack{\|x\|=1\\ x\in S_{k-1}^\perp}} x'Ax.$$

Corollary 2.1 ([8, Rayleigh-Quotient]) *Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a graph and $\mathcal{L}$ be the Laplacian of $\mathcal{G}$. We already know from the Perron-Frobenius lemma (Lemma 2.5) that the smallest eigenvalue is $\lambda_1 = 0$ with eigenvector $v_1 = \mathbf{1}$. By the Courant-Fischer Formula, we get*

$$\lambda_2(\mathcal{L}) = \min_{\substack{\|x\|=1\\ x\perp\mathbf{1}}} x'\mathcal{L}x.$$

Lemma 2.8 ([9, Laplacian Spectrum of a Graph Product]) *If $\mathcal{L}_G$ has eigenvalues $\lambda_1, \ldots, \lambda_n$ and $\mathcal{L}_H$ has eigenvalues $\mu_1, \ldots, \mu_n$, then $\mathcal{L}_{G\times H}$ has eigenvalues $\lambda_i + \mu_j, i, j = 1, \ldots, n$.*

Proof For $i = 1, 2, \ldots, n$, let us assume that $\mathcal{L}_G$ has eigenvalues λ_i with corresponding eigenvectors $X^{(i)}$ and $\mathcal{L}_H$ has eigenvalues μ_i with corresponding eigenvectors $Y^{(i)}$. Then, a simple multiplication shows that $\mathcal{L}_{G\times H}$ has for all $i, j \in [n]$, an eigenvector $Z^{(i,j)}$ of size $n^2 \times 1$ corresponding to an eigenvalue $\lambda_i + \mu_j$ ($Z^{(i,j)}$ has one entry for each possible pair of $(u, v) \in [n] \times [n]$), and such that its (u, v)th entry equals to $X_u^{(i)} Y_v^{(i)}$, i.e., $Z_{(u,v)}^{(i,j)} = X_u^{(i)} Y_v^{(i)}$. □

Lemma 2.9 *Suppose C is a stochastic matrix and $y = Cx$; then*

$$diam(conv(y)) \leq (1 - \mu(C))diam(conv(x)),$$

where $\mu(C) = \min_{i\neq j}(\sum_{k=1}^{n} \min(c_{ik}, c_{jk}))$. *In particular, when C is a scrambling matrix with* $\min^+ C \geq \delta$, *then we can say that* $\mu(C) \geq \delta$, *or* $diam(conv(y)) \leq (1 - \delta)diam(conv(x))$.

Proof A short proof of the above lemma can be found in [10]. □

2.4 Relevant Results on Game Theory

In this section, we briefly review some background material from game theory.

A finite noncooperative game $\Gamma = ([n], \{\mathcal{A}_i\}, \{U_i\})$ is defined in terms of a set of players $[n]$, a finite action set $\mathcal{A}_i$, and a utility function $U_i(a_i, a_{-i})$, $a \in \mathcal{A}_1 \times \cdots \times \mathcal{A}_n$, for each player $i \in [n]$. Note that in this definition the utility of player i is a function of its own action as well as those of others $a_{-i} \in (\mathcal{A}_1, \ldots, \mathcal{A}_{i-1}, \mathcal{A}_{i+1}, \mathcal{A}_n)$. We refer to the set of probability distributions over the action set $\mathcal{A}_i$ as the set of mixed strategies for player i.

Definition 2.2 Given an n-player game Γ, a strategies profile n-tuple $S = (\sigma_1^*, \sigma_2^*, \ldots, \sigma_n^*)$ is said to be a Nash equilibrium if

$$U_i(\sigma_i^*, \sigma_{-i}^*) \geq U_i(\sigma_i, \sigma_{-i}^*), \forall i \in [n], \sigma_i \in \Sigma_i,$$

where Σ_i and $U_i(\cdot)$ denote, respectively, the (possibly mixed) strategy set and utility[1] function of the ith player.

In other words, an n-tuple of strategies is said to be in equilibrium if no player has any positive incentive for changing his strategy, assuming that none of the other players is going to change its strategy. Next, we introduce a class of games known as *potential games* which can be considered as a subclass of games strategically equivalent to *team problems* [11].

[1] In this definition, we consider the players as maximizers whose goals are to maximize their individual utility functions. Analogously, one can consider the players as minimizers whose goals are to minimize their individual *cost* functions $C_i(\cdot)$. In such a case σ^* is said to be an NE if $C_i(\sigma_i^*, \sigma_{-i}^*) \leq C_i(\sigma_i, \sigma_{-i}^*)$, $\forall i \in [n]$, $\sigma_i \in \Sigma_i$.

Definition 2.3 Given an n-player game Γ, let $\mathcal{A} = \mathcal{A}_1 \times \cdots \times \mathcal{A}_n$ be the set of action profiles of the players. The game Γ is an ordinal potential game if there is a function $\Phi : \mathcal{A} \to \mathbb{R}$ such that $\forall a_{-i} \in \mathcal{A}_{-i},\ \forall \hat{a}_i,\ \tilde{a}_i \in \mathcal{A}_i$ we have

$$U_i(\hat{a}_i, a_{-i}) - U_i(\tilde{a}_i, a_{-i}) > 0 \Leftrightarrow \Phi(\hat{a}_i, a_{-i}) - \Phi(\tilde{a}_i, a_{-i}) > 0.$$

Intuitively, a game is said to be an ordinal potential game if the incentive of all players to change (improve) their strategy can be captured through a single global function called the potential function.

Theorem 2.1 ([12]) *Every finite ordinal potential game possesses a pure-strategy Nash equilibrium.*

Note that a pure-strategy equilibrium is an equilibrium over the action profiles of the players.

Definition 2.4 Given an n-player game with at least one pure-strategy Nash equilibrium over the finite action space $\mathcal{A} = \mathcal{A}_1 \times \cdots \times \mathcal{A}_n$, let $C_i(\cdot)$ denote player i's cost function, and $C : \mathcal{A} \to \mathbb{R}$ be a social cost function defined by $C(a) = \sum_{i=1}^{n} C_i(a)$. Then, the price of anarchy (PoA) of this game is defined to be

$$PoA = \frac{\max_{a \in \mathbf{NE}} C(a)}{\min_{a \in \mathcal{A}} C(a)},$$

where **NE** denotes the set of all the pure-strategy Nash equilibria of the game.

In fact, price of anarchy is a concept in game theory that measures how the efficiency of a system degrades due to selfish behavior of its agents [13].

Definition 2.5 Given a set Ω, a set function $f : 2^{\Omega} \to \mathbb{R}$ is called a sub-modular function if for any two subsets $A, B \subseteq \Omega$ with $A \subseteq B$ and any $x \in \Omega \backslash B$, we have

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

Sub-modular functions feature a natural diminishing returns property. In other words, the marginal return when a single element is added to an input set decreases as the size of the input set increases. An equivalent form of the above definition is to say that for any two sets $A, B \subseteq \Omega$, we have $f(A) + f(B) \geq f(A \cup B) + f(A \cap B)$.

2.5 Relevant Results on Probability Theory

In this section, we briefly review some background material from probability theory.

Definition 2.6 A martingale is a sequence $X_0, X_1, \ldots, X_m$ of random variables, where for $0 \leq i < m$, we have $\mathbb{E}[X_{i+1} | X_i, X_{i-1}, \ldots, X_0] = X_i$.

From the above definition it is easy to see that if $X_0, X_1, \ldots, X_m$ is a martingale, then $\mathbb{E}[X_i] = \mathbb{E}[X_0]$, for all $i = 1, 2 \ldots, m$.

Lemma 2.10 ([14, Azuma's Inequality]) *Let $X_0, X_1, \ldots, X_m$ be a martingale with $\mu := \mathbb{E}[X_0]$ and $|X_{i+1} - X_i| \le 1$ for all $0 \le i < m$. Then*

$$\mathbb{P}\Big(|X_m - \mu| > \sqrt{m}\theta\Big) \le 2e^{-\frac{\theta^2}{2}}.$$

Lemma 2.11 ([14, Tail formula for expectation]) *Suppose that N has a discrete distribution, taking values in $\mathbb{N}$. Then $\mathbb{E}[N] = \sum_{i=1}^{\infty} \mathbb{P}(N \ge i)$.*

Lemma 2.12 ([14, Chernoff Bound]) *Suppose $X_1, \ldots, X_n$ are independent random variables taking values in $\{0, 1\}$. Let X denote their sum and let $\mu = \mathbb{E}[X]$ denote the sum's expected value. Then for any $\delta > 0$,*

$$\mathbb{P}(X > (1+\delta)\mu) \le e^{-\frac{\delta^2\mu}{3}}$$
$$\mathbb{P}(X < (1-\delta)\mu) \le e^{-\frac{\delta^2\mu}{2}}.$$

References

1. P. Tetali, "Random walks and the effective resistance of networks," *Journal of Theoretical Probability*, vol. 4, pp. 101–109, 1991.
2. L. Lovász, "Random walks on graphs: A survey," *Combinatorics, Paul Erdos is eighty*, vol. 2, no. 1, pp. 1–46, 1993.
3. M. Catral, S. J. Kirkland, M. Neumann, and N.-S. Sze, "The Kemeny constant for finite homogeneous ergodic Markov chains," *Journal of Scientific Computing*, vol. 45, no. 1-3, pp. 151–166, 2010.
4. D. Aldous and J. Fill, *Reversible Markov chains and random walks on graphs*. Book in preparation - preprint available at http://www.stat.Berkeley.EDU/users/aldous, 2002.
5. D. A. Levin, Y. Peres, and E. L. Wilmer, *Markov Chains and Mixing Times*. American Mathematical Society, 2009.
6. L. Saloff-Coste, "Lectures on finite Markov chains," in *Lectures on Probability Theory and Statistics*. Springer, 1997, pp. 301–413.
7. C. D. Godsil, G. Royle, and C. Godsil, *Algebraic Graph Theory*. Springer, New York, 2001.
8. R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge University Press, 2012.
9. W. N. Anderson Jr and T. D. Morley, "Eigenvalues of the Laplacian of a graph*," *Linear and Multilinear Algebra*, vol. 18, no. 2, pp. 141–145, 1985.
10. J. Shen, "A geometric approach to ergodic non-homogeneous Markov chains," *Wavelet Analysis and Multiresolution Methods*, pp. 341–366, 2000.
11. T. Başar and G. J. Olsder, *Dynamic Noncooperative Game Theory*. SIAM, 1999, vol. 23.
12. D. Monderer and L. S. Shapley, "Potential games," *Games and Economic Behavior*, vol. 14, no. 1, pp. 124–143, 1996.
13. E. Koutsoupias and C. Papadimitriou, "Worst-case equilibria," in *STACS 99*. Springer, 1999, pp. 404–413.
14. N. Alon and J. H. Spencer, *The Probabilistic Method*. John Wiley & Sons, 2004.

Chapter 3
Unbiased Quantized Consensus

In this chapter, we describe the discrete-time unbiased quantized consensus model and the associated algorithm for computing the average values of the agents as introduced in [1]. In general, computing the average and quantized averages is known to be an important task in various contexts, such as multi-agent coordination and distributed averaging [2–4], e.g., when a set of robots coordinate in order to move to the same location; information fusion in sensor networks [5], e.g., when every sensor in a sensor network has a different measurement of the temperature and the sensors' common goal is to compute the average temperature; task assignment [6], e.g., when a group of agents has to reach a consensus on an optimal distribution of tasks, under communication and assignment constraints; decentralized voting [7], e.g., when nodes initially vote Yes or No and the goal is to find the majority opinion; and load balancing in processor networks, which has various applications in the operation of computer networks [8].

More specifically, we will consider a set of n "agents" or "nodes", each having initially an integer value in a certain range $[l, L]$. As we mentioned earlier, one of the reasons for considering integer values for the agents is that in many applications, due to limited memory and energy, agents' states are constrained to be discrete quantities, which leads to finite quantization of states. These nodes can communicate with their neighbors in a certain undirected, connected communication graph (which may be time-varying) and the goal of each agent is to compute the average of all initial values. The typical goal of computing the average must now be revised since the average may not be an integer. Thus we define a collection of integer values to belong to the "consensus set" if any two integers are at most one apart. We would like to design and analyze a protocol which brings the collection of integers at the nodes to the consensus set as fast as possible. In particular, in this chapter, the question of expected time to convergence for unbiased quantized consensus on undirected connected graphs is addressed.

S.R. Etesami, *Potential-Based Analysis of Social, Communication, and Distributed Networks*, Springer Theses, DOI 10.1007/978-3-319-54289-8_3

3.1 Unbiased Quantized Consensus Protocol

In this section, we provide a formal definition of the unbiased quantized consensus model for a fixed network as introduced in [1].

- There is a set of n agents, $\mathcal{V} = \{1, 2, \ldots, n\}$, which are connected on some undirected graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$.
- Each agent has an initial value $x_i(0) \in [l, L]$, which is a positive integer.
- At each time instant $t = 1, 2, \ldots$, one edge is chosen uniformly at random among the set of all the edges $\mathcal{E}$, and the incident nodes on the sides of this edge (let us call them i and j) update their values according to:

$$x_i(t+1) = \begin{cases} x_i(t) - 1, & \text{if } x_i(t) > x_j(t) \\ x_i(t) + 1, & \text{if } x_i(t) < x_j(t) \\ x_i(t), & \text{if } x_i(t) = x_j(t), \end{cases} \tag{3.1}$$

and the same holds for agent j. We refer to $x_i(t)$ as the *opinion of agent* i at time t and $x(t)$ as the *opinion profile* at time t.

3.1.1 Relevant Existing Results

First we note that, since the updating rules given in (3.1) preserve the average value, we have that under the unbiased quantized dynamics $\bar{x}(t) = \frac{\sum_{i=1}^{n} x_i(t)}{n}$ does not change with time and equals $\bar{x}(0)$ for all $t \geq 0$. Moreover, it has been shown earlier in [1] that such dynamics will converge with probability 1 to the consensus set,

$$C = \left\{ x | x_i \in \{\lfloor \bar{x}(0) \rfloor, \lfloor \bar{x}(0) \rfloor + 1\}, \ \frac{\sum_{i=1}^{n} x_i}{n} = \bar{x}(0), \ i \in [n] \right\},$$

and that the Lyapunov function defined by

$$V(t) = \sum_{i=1}^{n} \left(x_i(t) - \bar{x}(0) \right)^2 \tag{3.2}$$

will decrease by at least 2 after each *nontrivial* update. By nontrivial update we mean that the values of the incident nodes of the chosen edge (i, j) at time instant t differ by at least 2, i.e. $|x_i(t) - x_j(t)| \geq 2$. It was also shown in [1, Lemmas 4 and 5] that, after at most $\frac{(L-l)^2}{8} n$ nontrivial updates, the dynamics will terminate, where L and l denote, respectively, the maximum and minimum opinions at the beginning of the process. Thus a key part of the analysis of expected convergence time of such protocols involves bounding the maximum expected time until a nontrivial update happens.

For this purpose, let us define $\overline{T}$ as the maximum expected time such that a nontrivial update takes place over all possible configuration of integers. Note that in this maximization we exclude the configurations that are already in the consensus set, as otherwise our goal, i.e., reaching to the consensus, is achieved (no other nontrivial update can occur). To be more precise, let us introduce some notation for this quantity: for integers $j_1, \ldots, j_n$ define $T(j_1, \ldots, j_n)$ to be the expected time until a nontrivial update takes place when node i begins with integer value j_i. Then $\overline{T}$ is the largest possible $T(j_1, \ldots, j_n)$ for any configuration of integers which does not lie in the consensus set C:

$$\overline{T} = \max_{\substack{(j_1,\ldots,j_n)\in\mathbb{Z}^n, l\le j_i\le L \\ (j_1,\ldots,j_n)\notin C}} T(j_1, \ldots, j_n).$$

3.1.2 Random Walk Interpretation

In the above setting, let us for a moment assume that all the agents on the graph $\mathcal{G}$ have value 1 except two of them which are 0 and 2 (we will see soon that this assumption can be made without any loss of generality). At each time instant t, one edge will be selected with equal probability $\frac{1}{m}$ where m is the number of edges, and the incident nodes update their values based on (3.1). Therefore, we can interpret this problem in an alternative way. Consider two walkers, let us call them 0 and 2, who start a random walk on the vertices of the graph $\mathcal{G}$ whenever the selected edge is incident to at least one of them. To see this more clearly, let us consider a network of n nodes such that all of the nodes have value 1 except two of them which have values 0 and 2. Therefore, in the next update of the protocol (3.1), either the selected edge is incident to neither of the nodes with values 0 or 2, in which case there will not be any change, or the selected edge is incident to the node with value 0 and one of the nodes with value 1 (similarly to node with value 2 and one of the nodes with value 1). In this case 0 and 1 on the sides of the selected edge will be swapped (analogously, 2 and 1 on the sides of the selected edge will be swapped). This can be viewed as a random walk that the nodes with values 0 and 2 take to their next positions. Therefore, in this scenario $\overline{T}$ is equal to the maximum of the expected time it takes for these two walkers to meet.

3.2 Expected Convergence Time over Static Networks

In this section we start our expected convergence time analysis over static networks. But before we proceed, and based on the random walk interpretation we provided earlier, we consider the following definition.

Definition 3.1 Denoting the current locations of the walkers by x and y, if the selected edge at the next time instant is incident to one of the walkers, e.g., $\{x, x_i\}$ for some $x_i \in N(x)$, we will move that walker from node x to node x_i, otherwise the walkers will not change their positions. We refer to such a random walk process as the ***original*** process, and denote its expected meeting time by $M^{\mathrm{o}}(x, y)$.

Proposition 3.1 $\overline{T} = \max_{x,y \in V} \ M^{\mathrm{o}}(x, y)$.

Proof Let $\mathcal{S}$ denote the set of all the vectors of size n in which one entry is 0, one entry is 2, and the remaining entries are 1. It is immediate that $\overline{T} \geq \max_{(j_1,\ldots,j_n)\in\mathcal{S}} T(j_1, \ldots, j_n)$, since $\mathcal{S}$ is a subset of all the possible configurations. On the other hand, for any configuration of integer values $(j_1, j_2, \ldots, j_n) \notin C$ on the vertices of the graph $\mathcal{G}$, let us denote the maximum and minimum of these values by $j_{\max}$ and $j_{\min}$, respectively. Now let us consider a new configuration associated with $(j_1, j_2, \ldots, j_n)$ such that maps one of nodes with value $j_{\min}$ to 0, one of the nodes with value $j_{\max}$ to 2, and any other node to 1. Clearly the new configuration belongs to $\mathcal{S}$ such that if an update is a trivial update in configuration $(j_1, j_2, \ldots, j_n)$ it will be a trivial update for the new configuration as well, but not necessarily vice versa. This shows that for any possible configuration $(j_1, j_2, \ldots, j_n)$ there exists at least one correspondence in $\mathcal{S}$ such that the expected time for a non-trivial update in the corresponding configuration is at least as large as that in $(j_1, j_2, \ldots, j_n)$. Therefore, we have $\overline{T} \leq \max_{(j_1,\ldots,j_n)\in\mathcal{S}} T(j_1, \ldots, j_n)$, which together with the first inequality shows that $\overline{T} = \max_{(j_1,\ldots,j_n)\in\mathcal{S}} T(j_1, \ldots, j_n)$. Now consider any configuration in which all the agents on the graph $\mathcal{G}$ have value 1 except two of them which are 0 and 2. Focusing on the nodes which have a 0 and a 2, we see that they perform a random walk according to the original process, and a non-trivial update occurs at precisely the meeting time. □

One important fact is that both of the walkers in the original process have the same source of randomness, which selects an edge at each time instant. Therefore, these random walks are jointly correlated. In fact, in this section, we find an explicit form for $\overline{T}$ for a general fixed network. Now, let us consider the original process for two walkers on a finite graph. In order to compute $\overline{T}$, we introduce another process, called ***virtual*** process, to facilitate our analysis.

Definition 3.2 We define the virtual process to be the same as the original process until the time when the walkers become each other's neighbors, i.e. $x \in N(y)$, for some $x, y \in V$. At this time we count the connecting edge in the virtual process twice in our edge probability distribution. Moreover, we denote the meeting time function of the virtual process by $M^{\mathrm{v}}(x, y)$ for every two initial states x and y.

Note that, in the virtual process, as long as the walkers are not each other's neighbors, one edge of the network is picked, with probability $\frac{1}{m}$ and the movement of the walkers will be precisely as in the original process. But if the walkers are each other's neighbors, i.e. $x \in N(y)$, the edge selection probability slightly changes. In this case, the probability of selecting an edge $\mathbb{P}(e)$, $e \in \mathcal{E}(\mathcal{G})$ will be as follows:

$$\mathbb{P}(e) = \begin{cases} \frac{2}{m}, & \text{if } e = \{x, y\} \\ \frac{1}{m}, & \text{if } e \text{ is incident to either } x \text{ or } y \\ \frac{m-d(x)-d(y)}{m(m+1-d(x)-d(y))}, & \text{if } e \text{ is incident to neither } x \text{ nor } y, \end{cases}$$

and the walkers move depending on whether the selected edge is incident to them or not.

Remark 2 In order to have valid transition probabilities for the virtual process, we must have $d(x) + d(y) \le m, \forall (x, y) \in \mathcal{E}(\mathcal{G})$. This condition naturally holds for all connected graphs except the star graph and double-star graph, i.e., two star graphs whose centers are connected to each other (Figure A.1). In these cases we have $\max\{d(x) + d(y)|\ (x, y) \in \mathcal{E}(\mathcal{G})\} = m + 1$. However, the maximum expected meeting time of these two special cases can be computed precisely and directly without using a virtual process (Lemma A.1). Therefore, henceforth we assume that $d(x) + d(y) \le m, \forall (x, y) \in \mathcal{E}(\mathcal{G})$.

Given a graph $\mathcal{G}$, we choose an edge at random and uniformly among all the set of edges at each time instance. Let $\mathcal{Z}$ be the lazy random walk which is generated based on marginal distribution of the original process. In other words, the walker will move towards one of his neighbors with equal probability if he is located at one of the incident vertices. It is not hard to see that $\mathcal{Z}$ has the following transition probabilities:

$$P_{\mathcal{Z}}(x, y) = \begin{cases} 1 - \frac{d(x)}{m}, & \text{if } y = x \\ \frac{1}{m}, & \text{if } y \in N(x) \\ 0, & \text{else.} \end{cases} \tag{3.3}$$

Since the above transition matrix is doubly stochastic, $\pi = (\frac{1}{n}, \frac{1}{n}, \dots, \frac{1}{n})^T$ is its stationary distribution. This results in $\pi_i P(i, j) = \pi_j P(j, i)\ \forall i, j \in 1, \dots, n$, and hence $\mathcal{Z}$ is a reversible Markov chain.

We already know [9] that every reversible Markov chain has a hidden vertex w such that the hitting time from w to every state is less than or equal to the hitting time from that particular node to state w, i.e., w is a hidden vertex for $\mathcal{Z}$ if $H_{\mathcal{Z}}(w, x) \le H_{\mathcal{Z}}(x, w)\ \forall x$.

Definition 3.3 Assume that w is a hidden vertex for the reversible Markov chain $\mathcal{Z}$ with transition matrix given by (3.3). As in [9, 10], we define the potential function $\Phi(\cdot, \cdot) : \mathcal{V} \times \mathcal{V} \to \mathbb{R}$ to be

$$\Phi(x, y) = H_{\mathcal{Z}}(x, y) + H_{\mathcal{Z}}(y, w) - H_{\mathcal{Z}}(w, y).$$

We are now in a position to start our analysis. First, we briefly describe the stages that we will go through toward proving the result for general static networks. As discussed earlier, $\overline{T}$ is equal to the maximum expected meeting time of the original process. Since, due to the coupling between the random walks, computing the expected meeting time of the original process is difficult, we consider a virtual process which closely follows the original process until the time when the walkers are at certain locations (more precisely when they are each other's neighbors). In fact, one can think of the virtual process as an approximated process for the original one. However, the virtual process is itself a jointly correlated random walk. Therefore, to characterize its expected meeting time function, i.e., $M^{\mathrm{v}}(x, y)$, we will show that $M^{\mathrm{v}}(x, y)$ follows almost the same recursion formula as $\Phi(x, y)$. This allows us to construct a harmonic function (Lemma 3.1) using $\Phi(x, y)$ and $M^{\mathrm{v}}(x, y)$. We show that such a harmonic function is zero at some boundary point, and hence, must be identical to zero. This allows us to characterize $M^{\mathrm{v}}(x, y)$ based on $\Phi(x, y)$ (Theorem 3.2). Furthermore, since $\Phi(x, y)$ is a function of the expected hitting time of a single lazy random walk $\mathcal{Z}$ (with no further coupling), we can find an expression for $M^{\mathrm{v}}(x, y)$ based on only the expected hitting time functions of the lazy random walk $\mathcal{Z}$. Moreover, since such an expression does not involve any coupling term, it is easy to compute it for different networks. Finally, we show in Theorem 3.3 that the expected meeting time function of the virtual process $M^{\mathrm{v}}(x, y)$, and that in the original process $M^{\mathrm{o}}(x, y)$, lie within a constant factor of each other. This establishes our tight bound for $\overline{T}$. We now start the stages of our proof, with the following lemma.

Lemma 3.1 *A function $f : \mathcal{V} \times \mathcal{V} \to \mathbb{R}$ defined by $f(x, y) = \frac{1}{2}\Phi(x, y) - M^{\mathrm{v}}(x, y)$ is harmonic for the simple random walk on $\mathcal{G} \times \mathcal{G}$, i.e.*

$$f(x, y) = \sum_{(r,s)\in\mathcal{V}\times\mathcal{V}} \mathcal{Q}\big((x, y), (r, s)\big) f(r, s),$$

where $\mathcal{Q}$ is the transition matrix of the simple random walk on $\mathcal{G} \times \mathcal{G}$, i.e.

$$\mathcal{Q}\big((x, y), (r, s)\big) = \begin{cases} \frac{1}{d(x)+d(y)}, & \textit{if } (r, s) \in N_{\mathcal{G}\times\mathcal{G}}(x, y) \\ 0, & \textit{else}. \end{cases}$$

Proof By the transitivity property of reversible Markov chains, we note that $\Phi(x, y)$ is symmetric, i.e. for any hidden vertex w,

$$\begin{aligned} \Phi(x, y) &= H_{\mathcal{Z}}(x, y) + H_{\mathcal{Z}}(y, w) - H_{\mathcal{Z}}(w, y) \\ &= H_{\mathcal{Z}}(y, x) + H_{\mathcal{Z}}(x, w) - H_{\mathcal{Z}}(w, x) = \Phi(y, x). \end{aligned}$$

Therefore, we can write:

$$\begin{aligned} \Phi(x, y) &= \frac{d(x)}{d(x) + d(y)}\big[H_{\mathcal{Z}}(x, y) + H_{\mathcal{Z}}(y, w) - H_{\mathcal{Z}}(w, y)\big] \\ &\quad + \frac{d(y)}{d(x) + d(y)}\big[H_{\mathcal{Z}}(y, x) + H_{\mathcal{Z}}(x, w) - H_{\mathcal{Z}}(w, x)\big] \\ &= \frac{d(x)}{d(x) + d(y)}\big(H_{\mathcal{Z}}(y, w) - H_{\mathcal{Z}}(w, y)\big) \end{aligned}$$

$$+\frac{d(y)}{d(x)+d(y)}\big(H_{\mathcal{Z}}(x,w)-H_{\mathcal{Z}}(w,x)\big)$$
$$+\Big[\frac{d(x)}{d(x)+d(y)}H_{\mathcal{Z}}(x,y)+\frac{d(y)}{d(x)+d(y)}H_{\mathcal{Z}}(y,x)\Big]. \tag{3.4}$$

Also, by expanding $H_{\mathcal{Z}}(x,y)$ by one step, we get:

$$H_{\mathcal{Z}}(x,y)=\frac{m}{d(x)}+\frac{1}{d(x)}\sum_{j\in N(x)}H_{\mathcal{Z}}(j,y), \tag{3.5}$$

and similarly by switching x and y we have:

$$H_{\mathcal{Z}}(y,x)=\frac{m}{d(y)}+\frac{1}{d(y)}\sum_{j\in N(y)}H_{\mathcal{Z}}(j,x). \tag{3.6}$$

Using (3.5) and (3.6) in (3.4), we get

$$\begin{aligned}\Phi(x,y)=\frac{2m}{d(x)+d(y)}&+\frac{d(x)}{d(x)+d(y)}\big(H_{\mathcal{Z}}(y,w)-H_{\mathcal{Z}}(w,y)\big)\\&+\frac{d(y)}{d(x)+d(y)}\big(H_{\mathcal{Z}}(x,w)-H_{\mathcal{Z}}(w,x)\big)\\&+\frac{1}{d(x)+d(y)}\left(\sum_{j\in N(x)}H_{\mathcal{Z}}(j,y)+\sum_{j\in N(y)}H_{\mathcal{Z}}(j,x)\right).\end{aligned} \tag{3.7}$$

Also, from the definition of $\Phi(\cdot,\cdot)$, we have $\Phi(j,y)=H_{\mathcal{Z}}(j,y)+H_{\mathcal{Z}}(y,w)-H_{\mathcal{Z}}(w,y),\ \forall j\in N(x)$. By taking summation over all $j\in N(x)$ and multiplying by the factor $\frac{1}{d(x)+d(y)}$, we arrive at

$$\begin{aligned}\frac{1}{d(x)+d(y)}\sum_{j\in N(x)}\Phi(j,y)=&\frac{1}{d(x)+d(y)}\sum_{j\in N(x)}H_{\mathcal{Z}}(j,y)\\&+\frac{d(x)}{d(x)+d(y)}\big[H_{\mathcal{Z}}(y,w)-H_{\mathcal{Z}}(w,y)\big].\end{aligned} \tag{3.8}$$

By the same argument, and since $\Phi(x,j)=H_{\mathcal{Z}}(j,x)+H_{\mathcal{Z}}(x,w)-H_{\mathcal{Z}}(w,x)$, we have

$$\begin{aligned}\frac{1}{d(x)+d(y)}\sum_{j\in N(y)}\Phi(j,x)=&\frac{1}{d(x)+d(y)}\sum_{j\in N(y)}H_{\mathcal{Z}}(j,x)\\&+\frac{d(y)}{d(x)+d(y)}\big[H_{\mathcal{Z}}(x,w)-H_{\mathcal{Z}}(w,x)\big].\end{aligned} \tag{3.9}$$

Substituting (3.9) and (3.8) in (3.7) gives us

$$\Phi(x, y) = \frac{2m}{d(x) + d(y)} + \frac{1}{d(x) + d(y)} \left(\sum_{j \in N(x)} \Phi(j, y) + \sum_{j \in N(y)} \Phi(j, x) \right). \tag{3.10}$$

On the other hand, we note that regardless of whether $y \notin N(x)$ or $y \in N(x)$, by expanding the meeting time of the virtual process one step we have

$$M^{\text{v}}(x, y) = \left(1 - \frac{d(x) + d(y)}{m}\right) \left(1 + M^{\text{v}}(x, y)\right) + \sum_{j \in N(x)} \frac{1}{m} \left(1 + M^{\text{v}}(j, y)\right) + \sum_{j \in N(y)} \frac{1}{m} \left(1 + M^{\text{v}}(j, x)\right)$$

from which by simplifying and rearranging the terms we get

$$M^{\text{v}}(x, y) = \frac{m}{d(x) + d(y)} + \frac{1}{d(x) + d(y)} \left(\sum_{j \in N(x)} M^{\text{v}}(j, y) + \sum_{j \in N(y)} M^{\text{v}}(j, x) \right). \tag{3.11}$$

Let $S(x, y) := \frac{\Phi(x, y)}{2}$. From (3.10) it is not hard to see that

$$S(x, y) = \frac{m}{d(x) + d(y)} + \frac{1}{d(x) + d(y)} \left(\sum_{j \in N(x)} S(j, y) + \sum_{j \in N(y)} S(j, x) \right). \tag{3.12}$$

We consider the simple random walk $\mathcal{Q}$ on the Cartesian product graph $\mathcal{G} \times \mathcal{G}$. The cover time and hitting time of such graphs have been extensively studied in [11–13]. We show that the function $f(x, y) = S(x, y) - M^{\text{v}}(x, y)$ is harmonic on $\mathcal{G} \times \mathcal{G}$ for the transition matrix $\mathcal{Q}$. In fact, from (3.11) and (3.12) we have

$$\begin{aligned} f(x, y) &= \frac{1}{d(x) + d(y)} \sum_{j \in N(x)} \left(S(j, y) - M^{\text{v}}(j, y)\right) + \frac{1}{d(x) + d(y)} \sum_{j \in N(y)} \left(S(j, x) - M^{\text{v}}(j, x)\right) \\ &= \frac{1}{d(x) + d(y)} \left(\sum_{j \in N(x)} f(j, y) + \sum_{j \in N(y)} f(x, j) \right) \\ &= \sum_{(r,s) \in \mathcal{V} \times \mathcal{V}} \mathcal{Q}\big((x, y), (r, s)\big) f(r, s). \end{aligned}$$

This completes the proof. □

Now we are ready to characterize the expected meeting time of the virtual process based on the expected hitting times of the single lazy random walk $\mathcal{Z}$.

Theorem 3.2 *The expected meeting time of the virtual process initiated from* x, y *is equal to* $\frac{1}{2}\Phi(x, y)$*, i.e.,*

$$M^{\text{v}}(x, y) = \frac{1}{2}\Big[H_{\mathcal{Z}}(x, y) + H_{\mathcal{Z}}(y, w) - H_{\mathcal{Z}}(w, y)\Big].$$

Proof Let $g : \mathcal{V} \times \mathcal{V} \to \mathbb{R}$ be the zero function, i.e., $g \equiv 0$. Clearly, g is a harmonic function over $\mathcal{G} \times \mathcal{G}$. On the other hand, we have

$$\begin{aligned} f(w, w) &= S(w, w) - M^{\text{v}}(w, w) = S(w, w) \\ &= \frac{1}{2}\Big(H_{\mathcal{Z}}(w, w) + H_{\mathcal{Z}}(w, w) - H_{\mathcal{Z}}(w, w)\Big) = 0 = g(w, w). \end{aligned}$$

Since f and g are both harmonic functions for the transition matrix $\mathcal{Q}$ and also they have the same value at the node (w, w), using Lemma 2.4 they must be equal. Thus $f \equiv 0$, which shows that $M^{\text{v}}(x, y) = \frac{1}{2}\Phi(x, y), \forall x, y$. □

Theorem 3.3 *Consider a network* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$*. Then,*

$$\max_{x,y} M^{\text{v}}(x, y) \le \overline{T} \le 2 \max_{x,y} M^{\text{v}}(x, y).$$

Proof Initiating from arbitrary nodes x and y, we note that both the virtual process and the original process follow the same joint distribution until they are each other's neighbor. However, when the walkers are each other's neighbor, with higher probability they are going to meet in the virtual process than in the original process. Therefore, $M^{\text{v}}(x, y) \le \overline{T}$ for all x, y. Since by definition $\overline{T}$ is independent of the initial states of walkers, $\max_{x,y} M^{\text{v}}(x, y) \le \overline{T}$. For the upper bound, we use the same argument as in [10]. Again, as mentioned earlier, the virtual process and the original process remain the same until the two walkers become each other's neighbors, i.e. for some x, y with $x \in N(y)$. At this time, the probability that two walkers meet in the next transition in the original process is $\frac{1}{m}$, while this probability for the virtual process is $\frac{2}{m}$. Since the former is half of the latter, this immediately implies that the meeting time of the former is within a constant factor of the latter. In fact, at each time that the walkers in the virtual process meet, with probability $\frac{1}{2}$ the walkers in the original process meet as well. However, if the walkers in the virtual process meet, but they do not meet in the original process (which happens with probability $\frac{1}{2}$), then in the original process we may assume that the positions of the walkers have not changed, and in the virtual process we may assume that they just switch their positions (from x, y to y, x). Therefore, in this case we may assume that a new original process which is followed by its corresponding virtual process has just been initiated from nodes x and y. Since each of these collisions of walkers during different time intervals happens independently and with probability $\frac{1}{2}$, we can write

$$\overline{T} \leq \sum_{k=1}^{\infty} (\frac{1}{2})^k k \max_{x,y} M^{\mathrm{v}}(x, y) = 2 \max_{x,y} M^{\mathrm{v}}(x, y),$$

where in the above summation the term $(\frac{1}{2})^k$ corresponds to the probability that the walkers in the virtual process meet k times while they do not meet in the original process, and the term $k \max_{x,y} M^{\mathrm{v}}(x, y)$ is an upper bound for the expected time that the walkers in the virtual process meet k times. □

Next, we will proceed by computing $H_{\mathcal{Z}}(x, y)$. Note that if at the time instant t the walker is at the node $\mathcal{Z}(t)$, then by (3.3) the probability of staying in that state is $1 - \frac{d(\mathcal{Z}(t))}{m}$, where $d(\mathcal{Z}(t))$ denotes the degree of the node $\mathcal{Z}(t)$ where the walker at time t is located. Because of the Markov property of the random walk, the probability of moving out from each state follows the geometric distribution. Therefore, the expected time that the walker waits in state $\mathcal{Z}(t)$ is $\frac{m}{d(\mathcal{Z}(t))}$. However, when the walker moves to the next state, he will see all of his neighbors with the same probability. Therefore, the random walk $\mathcal{Z}(t)$ can be viewed as a simple random walk where the waiting time between its consecutive transitions given that the walk is at node $\mathcal{Z}(t)$ is equal to $\frac{m}{d(\mathcal{Z}(t))}$. Since every sample path of the random walk $\mathcal{Z}$ is equivalent to those of a simple random walk except that it has the expected waiting time of $\frac{m}{d(\mathcal{Z}(t))}$ between its consecutive transitions (note that this waiting time for simple random walk equals 1 for all the nodes), thus its hitting time $H_{\mathcal{Z}}(x, y)$ is equal to

$$\begin{aligned} H_{\mathcal{Z}}(x, y) &= \sum_i \mathbb{E}\begin{bmatrix}\text{number of visits to node } i \text{ in a} \\ \text{simple random walk before } \tau_y^x\end{bmatrix} \times \mathbb{E}\begin{bmatrix}\text{waiting time} \\ \text{at node } i\end{bmatrix} \\ &= \sum_i G_{\tau_y^x}(i) \times \frac{m}{d(i)} = m \sum_i \frac{G_{\tau_y^x}(i)}{d(i)} \\ &= \sum_i \frac{m}{2}\big[\mathcal{R}\big(x \leftrightarrow y\big) + \mathcal{R}\big(y \leftrightarrow i\big) - \mathcal{R}\big(x \leftrightarrow i\big)\big], \end{aligned} \tag{3.13}$$

where the last equality is due to Lemma 2.1.

Lemma 3.2 *Consider a connected network $\mathcal{G}$ with n nodes and m edges and diameter D. Then, for unbiased quantized consensus, we have*

$$\frac{1}{2} H_{\mathcal{Z}} \leq \overline{T} \leq 2H_{\mathcal{Z}} \leq 2nmD,$$

where, $H_{\mathcal{Z}} = \max_{x,y} H_{\mathcal{Z}}(x, y)$.

Proof Since w was a hidden vertex, we get $H_{\mathcal{Z}}(y, w) - H_{\mathcal{Z}}(w, y) \geq 0, \forall y$. Applying this inequality in the expression for $M^{\mathrm{v}}(x, y)$ in Theorem 3.2, and using the definition of $H_{\mathcal{Z}}$ we get

$$\frac{1}{2}H_{\mathcal{Z}}(x, y) \leq M^{\mathrm{v}}(x, y) \leq H_{\mathcal{Z}}, \forall x, y. \tag{3.14}$$

Using (3.14) and Theorem 3.3, we can see that $\frac{1}{2}H_{\mathcal{Z}} \leq \overline{T} \leq 2H_{\mathcal{Z}}$. Furthermore, by relation (3.13), we get

$$2H_{\mathcal{Z}} \leq \max_{x,y} \Big\{ m \sum_{i} \big[\mathcal{R}\big(x \leftrightarrow y\big) + \mathcal{R}\big(y \leftrightarrow i\big) - \mathcal{R}\big(x \leftrightarrow i\big) \big] \Big\} \leq 2nmD, \tag{3.15}$$

where, in the second inequality, we have used the fact that the effective resistance between any two nodes cannot exceed the length of the shortest path between those nodes [14] which is upper bounded by the diameter of $\mathcal{G}$. □

We are now ready to state our upper bound for the expected convergence time of the unbiased quantized consensus over static networks.

Theorem 3.4 *The expected time until the unbiased quantized dynamics reach a consensus set is* $O(nmD \log n)$.

Proof Consider the time it takes for the Lyapunov function $V(t) = \max_i x_i(t) - \min_i x_i(t)$ to shrink by at least 1 starting from any non-consensus configuration. We will next argue that this time is $O(nmD \log n)$, which will prove the statement of the theorem.

Indeed, let $S_{\max}(t) = \arg\max_i x_i(t)$, $S_{\min}(t) = \arg\min_i x_i(t)$. Now let us consider the event that $V(t') = V(t)$ for $t' > t$. We claim that there must exist $i \in S_{\max}(t)$, $j \in S_{\min}(t)$ such that two random walkers in the original process starting from nodes i and j have not yet met during the time interval $[t, t']$. To see this, we note that during the executions of the unbiased quantized dynamics no new values $\max_i x_i(t)$ and $\min_i x_i(t)$ will be created; these values only travel from one node to the other over the network based on the random walks induced by the original process. Now given that at time t' we have $V(t') = V(t)$, there must exist two nodes i', j' such that $x_{i'}(t') = \max_i x_i(t)$ and $x_{j'}(t') = \min_i x_i(t)$. By tracking back the origin of the random walks which bring the values $\max_i x_i(t)$ and $\min_i x_i(t)$ at time t to the nodes i' and j' at time t', one can see that these walks never meet (otherwise, they participate in a non-trivial update at some intermediate time and the values change to $\max_i x_i(t) - 1$ and $\min_i x_i(t) + 1$), and they must be originated from some nodes $i \in S_{\max}(t)$, $j \in S_{\min}(t)$. This shows that the probability that $V(t') = V(t)$ for $t' > t$ is upper bounded by the probability that there exist some $i \in S_{\max}(t)$, $j \in S_{\min}(t)$ such that two random walkers in the original process starting from nodes i and j have not yet met.

Thus letting $\mathcal{C}$ be the time elapsed until $V(t)$ shrinks by 1, and denoting $\tau^{\mathrm{o}}(x, y)$ as the meeting time of two random walkers in the original process starting from x and y, we have $\overline{T} = \max_{x,y} M^{\mathrm{o}}(x, y) = \max_{x,y} \mathbb{E}(\tau^{\mathrm{o}}(x, y))$. Therefore, we can write

$$\begin{aligned}
\mathbb{E}(\mathcal{C}) &= \sum_{t=0}^{\infty} \mathbb{P}(\mathcal{C} > t) \\
&\leq \sum_{t=0}^{\infty} \min \left\{ 1, \sum_{x,y} \mathbb{P}(\tau^{\mathrm{o}}(x, y) > t) \right\} \\
&\leq 2 \int_0^{\infty} \min \left\{ 1, n^2 e^{-\lfloor \frac{t}{e\overline{\overline{T}}} \rfloor} \right\} dt \\
&\leq 2 \int_0^{\infty} \min \left\{ 1, n^2 e^{(1-\frac{t}{2enmD})} \right\} dt \\
&= O\left(nmD \log n \right), \qquad (3.16)
\end{aligned}$$

where the equality is due to the tail formula for expectation, the first inequality is due to the union bound, the second inequality is due to Lemma 2.3 and upper bounding the sum with twice of its integral, the third inequality is valid due to Lemma 3.2, and the last equality follows from

$$\int_0^{\infty} \min \left\{ 1, Ae^{-at} \right\} dt = \frac{1 + \log A}{a},$$

which holds when $A \geq 1$, and is from [15, Chap. 5.3.2.]. □

3.2.1 Simulation Results

In this section we present some simulation results to provide a comparison between the maximum expected meeting time $\overline{\overline{T}}$ and the proposed upper and lower bounds given above. We consider four different types of graphs with n nodes: line graph, star graph, lollipop graph, and semi-regular graph. In lollipop graph each of its side clusters has $[\frac{n}{4}]$ nodes and they are connected with a single path. Also, for the semi-regular graph we consider a graph with n nodes arranged around a circle, such that each node is linked to its next four nodes when we move clockwise around the circle. In Fig. 3.1, the ratio $\frac{\overline{\overline{T}}}{mnD}$ is depicted for each graph. It can be seen that this ratio for the line graph converges asymptotically to a constant as n goes to infinity. Moreover, using the transition probabilities given in (3.3) for the star graph, a simple calculation shows that $H_{\mathcal{Z}} = \frac{n(n-1)}{2}$, and hence from Lemma 3.2 we get $\frac{n(n-1)}{4n^2} \leq \frac{\overline{\overline{T}}}{mnD} \leq \frac{n(n-1)}{n^2}$, which is consistent with the ratio given in Fig. 3.1. Finally, for lollipop and semi-regular graphs, although the ratio $\frac{\overline{\overline{T}}}{mnD}$ is oscillating, it is clearly bounded from above by 1, which confirms the upper bound provided in Lemma 3.2.

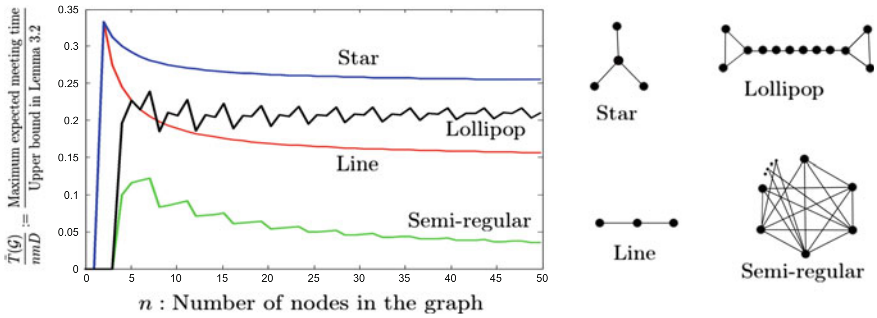


Fig. 3.1 Comparison between the maximum meeting time of the original process $\overline{T}$ and given upper bound mnD four different types of graphs

3.3 Expected Convergence Time over Time-Varying Networks

In this section we extend the results of the previous section to time-varying networks.

3.3.1 Unbiased Quantized Model over Time-Varying Networks

Let us consider a sequence of connected networks $\mathcal{G}(t) = (\mathcal{V}, \mathcal{E}(t))$, $t = 0, 1, 2, \ldots$, over a set of n vertices, $\mathcal{V} = \{1, 2, \ldots, n\}$. We assume that such a sequence is selected a priori arbitrarily, and then fixed at the beginning of the process. In particular, the future sequences of the graphs do not depend on the current states of the quantized consensus protocol which we will define later. Now, one can define unbiased quantized consensus over time-varying graphs $\{\mathcal{G}(t)\}$ to be very naturally the same as that for the static graphs, repeated for different graphs at different time instances. In other words, at each time instant $t = 0, 1, \ldots$, we select an edge uniformly at random from $\mathcal{E}(t)$ and let its incident nodes update their values based on the rule given by (3.1). Therefore, in the remainder of this chapter our goal is to obtain an upper bound for the expected convergence time of unbiased quantized consensus protocols over the sequence of time-varying networks $\{\mathcal{G}(t)\}$.

For this purpose, let us denote the number of edges and the degree of vertex x in $\mathcal{G}(t)$ by m_t and $d_t(x)$, respectively. Here, we assume that all the networks are connected, as otherwise we may not reach consensus through the dynamics. For example if the networks are allowed to be disconnected, one can easily find a sequence of networks such that the quantized algorithm does not converge to any specific outcome. Moreover, as in the case of static networks, and in order to be able to define a virtual process (Remark 2), we invoke the following assumption.

Assumption 1 $\mathcal{G}(t) = (\mathcal{V}, \mathcal{E}(t))$ are connected graphs such that $d_t(u) + d_t(v) \leq m_t, \forall (u, v) \in \mathcal{E}(t), \forall t = 0, 1, 2, \ldots$.

Note that the above assumption is a very mild one and simply states that the sequence of graphs can be arbitrarily chosen from the set of all connected graphs except star and double-star graphs. Under this assumption we analyze the expected convergence time of the dynamics.

3.3.2 Preliminary Definitions and Relevant Results

Let us define a sequence of lazy random walks $\{\mathcal{Z}_t\}_{t\geq 0}$ corresponding to each network with the following transition probability matrices:

$$P_t(x, y) = \begin{cases} 1 - \frac{d_t(x)}{m_t}, & \text{if } y = x \\ \frac{1}{m_t}, & \text{if } y \in N_t(x) \\ 0, & \text{else,} \end{cases} \tag{3.17}$$

where $N_t(x)$ denotes the set of neighbors of node x in $\mathcal{G}(t)$. Note that since all the probability matrices $P_t, t \geq 0$ are doubly stochastic, any arbitrary product is also doubly stochastic, and hence they all share a common stationary distribution $\pi = (\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n})$. Now based on the above setting and very naturally, one can extend the unbiased quantized protocol over the sequences of time-varying graphs. To be more precise, we consider similar dynamics over these sequences of networks as follows. At each time step $t = 0, 1, 2, \ldots$, we choose an edge uniformly at random from $\mathcal{G}(t)$ and update its incident nodes as in the static case. We note that since such dynamics preserve the average of the opinions over time-varying networks, the function given in (3.2) is a valid Lyapunov function for the dynamics over time-varying networks. In particular, the value of this function will decrease by at least 2 after every nontrivial update. Consequently, finding an upper bound on the maximum expected convergence time of the dynamics over time-varying networks reduces to that of finding the maximum expected time it takes for a nontrivial update to take place, which by an abuse of notation we denote by $\overline{T}$.

Next, we consider two random walkers who move jointly over these sequences of time-varying networks based on whether the selected edge is incident to them or not. In other words, if the chosen edge at some time instant is incident to one of the walkers, we will then move it, otherwise the walkers will not change their positions. Through some manipulation as in the case of static network analysis, one can argue that $\overline{T}$ is equal to the maximum expected time it takes for these two walkers to meet. Therefore, very similar to the static case, we have the following definitions.

Definition 3.4 Denoting the locations of the walkers at time instant t by $x, y \in \mathcal{V}(\mathcal{G}(t))$, if the selected edge at this time is incident to one of the walkers, e.g., $\{x, x_i\}$

for some $x_i \in N_t(x)$, we will move it from node x to node x_i, otherwise the walkers will not change their positions. We refer to such a random walk over the sequence of $\mathcal{G}(t), t = 0, 1, 2, \ldots$ as the original process.

Definition 3.5 In a time-varying network, a virtual process is the same as the original process until when the walkers are each other's neighbors at some time instant t, i.e. $x \in N_t(y)$, for some $x, y \in \mathcal{V}(\mathcal{G}(t))$. At this time the connecting edge $\{x, y\} \in \mathcal{E}(t)$ in the virtual process is counted twice in the edge probability distribution, i.e., for $e \in \mathcal{E}(t)$,

$$\mathbb{P}(e) = \begin{cases} \frac{2}{m_t}, & \text{if } e = \{x, y\} \\ \frac{1}{m_t}, & \text{if } e \text{ is incident to either } x \text{ or } y \text{ on } \mathcal{G}(t) \\ \frac{m_t - d_t(x) - d_t(y)}{m_t(m_t + 1 - d_t(x) - d_t(y))}, & \text{if } e \text{ is incident to neither } x \text{ nor } y \text{ on } \mathcal{G}(t), \end{cases}$$

and the walkers move depending on whether the selected edge is incident them or not.

Remark 3 Due to the time-varying nature of the networks, there is no dependency between the location of the walkers and the next graph in the sequence. In other words, the next graph in the sequence cannot be determined based on the current locations of the walkers, as otherwise, one can simply construct a sequence of connected time-varying networks which depends on the location of the walkers, such that the walkers never meet each other. We will see later that due to some laziness that exists in the joint transition probability of these random walks, the expected time until they meet is finite.

3.3.3 Expected Convergence Time over Time-Varying Networks

Let us consider the virtual process and denote its transition probabilities described on the network $\mathcal{G}(t) \times \mathcal{G}(t)$ by a matrix $K(t)$. In fact, $K(t)$ is an $n^2 \times n^2$ dimensional matrix whose rows and columns are labeled as all the possible pairs of vertices, and the entry $((x_1, y_1), (x_2, y_2))$ of $K(t)$ is the conditional probability that the walkers in the virtual process are at the nodes (x_2, y_2) given that they were at (x_1, y_1) at the previous time step. Based on this construction, the meeting time of the virtual process that started at (a, b) is equal to the expected time that a time inhomogeneous Markov chain with transition matrices $K(t), t = 0, 1, 2, \ldots$ started from (a, b) hits one of the states $S = \{(1, 1), (2, 2), \ldots, (n, n)\}$ for the first time. In fact, since we are interested in an upper bound on the expected hitting time of such a random walk on $\{\mathcal{G}(t) \times \mathcal{G}(t)\}_{t\geq 0}$ (and hence an upper bound on the expected meeting time of the virtual process), we can manipulate some of the entries of the matrices $\{K(t)\}_{t\geq 0}$ as long as we make sure that the expected hitting time to S does not decrease throughout the process. Therefore, we upper bound the expected hitting time of such a modified process whose transition matrix at time t we denote by $\bar{K}(t)$. Following this idea and in order to have a symmetric modified chain, for all $t \geq 0$ we define $K(t)$ to be the same as $\bar{K}(t)$ in all but the following cases:

- We note that in the virtual process when two walkers are each other's neighbors ($x \in N_t(y)$), the probability that the connecting edge between them is chosen is $\frac{2}{m_t}$. Therefore, in the modified chain matrix $\bar{K}(t)$ by assigning probabilities $\frac{1}{m_t}$ for moving from (x, y) to (x, x) and also moving from (x, y) to (y, y) (similarly moving from (y, x) to (x, x) and also moving from (y, x) to (y, y)), the expected hitting time to S will not change.
- Since all the vertices $(x, x) \in S$ are absorbing states in the virtual process, we have $K(t)((x, x), (x, x)) = 1, \forall x \in \mathcal{V}$. In this case, by modifying the row (x, x) of the matrix $K(t)$ to $\bar{K}(t)((x, x), (x, x)) = 1 - \frac{2d_t(x)}{m_t}$ and $\bar{K}(t)((x, x), (x', y')) = \frac{1}{m_t}, \forall (x', y') \in N_{\mathcal{G}(t)\times\mathcal{G}(t)}(x, x)$ we will get a chain whose expected hitting time to S is again the same as the expected hitting time of the chain $\{K(t)\}_{t\geq 0}$.

By these modifications, the modified chains $\{\bar{K}(t)\}_{t\geq 0}$, and $\{K(t)\}_{t\geq 0}$ will have the same expected hitting times to S. Moreover, by the definition of the transition probabilities of the virtual process matrices $(K(t))$ and the above modifications, we observe that the transition matrix $\bar{K}(t)$ must satisfy $\bar{K}(t) = I - \frac{1}{m_t}\mathcal{L}_{\mathcal{G}(t)\times\mathcal{G}(t)}$ for all $t \geq 0$, where $\mathcal{L}_{\mathcal{G}(t)\times\mathcal{G}(t)}$ is the Laplacian of the Cartesian product graph $\mathcal{G}(t) \times \mathcal{G}(t)$, and I denotes the identity matrix of proper size. On the other side by a close look at the matrix P_t it is not hard to see that $P_t = I - \frac{1}{m_t}\mathcal{L}_{\mathcal{G}(t)}$.

We are now in a position to study the expected convergence time of the dynamics. But, before we proceed, we first provide a summary of the steps involved in the proof. Based on the above discussion, in order to determine the expected meeting time function of the virtual process over $\{\mathcal{G}(t)\}_{t=0}^{\infty}$, we can equivalently concentrate on finding the expected hitting time to the absorbing states S of an inhomogeneous Markov chain with transition matrices $\{K(t)\}_{t=0}^{\infty}$ which are defined over $\{\mathcal{G}(t) \times \mathcal{G}(t)\}_{t=0}^{\infty}$. As discussed above, the hitting time to the absorbing states of this chain is equal to that in the modified chain $\{\bar{K}(t)\}_{t=0}^{\infty}$. Since the matrices $\{\bar{K}(t)\}_{t=0}^{\infty}$ are symmetric, and hence, doubly stochastic, we can find a precise expression for the second largest eigenvalue and the smallest eigenvalue of $\bar{K}(t)$ based on those of the matrix P_t. This allows us to find tight bounds on the spectral gap of the matrices $\bar{K}(t), t = 0, 1, \ldots$.

Since all matrices in the inhomogeneous Markov chain $\{\bar{K}(t)\}_{t=0}^{\infty}$ are doubly stochastic, starting from any initial distribution $p(0)$, and after sufficiently long period of time t' (which will be determined by the spectral gap of such matrices), the probability of being in different states $p(t')$ will be very close to the stationary distribution of the chain, i.e., $\pi = (\frac{1}{n^2}, \frac{1}{n^2}, \ldots, \frac{1}{n^2})'$. In particular, the probability of being absorbed by S after time t' will be large enough and bounded away from 0. This allows us to find an upper bound on how long it takes for the chain $\{\bar{K}(t)\}_{t=0}^{\infty}$ starting from an arbitrary initial distribution $p(0)$ to hit at least one of the absorbing states. Equivalently, this provides an upper bound on the expected hitting time of the chain $\{K(t)\}_{t=0}^{\infty}$ to the set S, and hence, an upper bound on the expected meeting time of the virtual process. Finally, by the same line of argument as in the case of static networks, we can show that the maximum expected meeting time of the original process is within a constant factor of that in the virtual process. Keeping these main steps in mind and toward a complete proof, we first consider the following lemma.

Now let us denote the second largest eigenvalue and the second smallest eigenvalue of a $k \times k$ matrix A by $\alpha_2(A)$ and $\alpha_{k-1}(A)$, respectively. For every $t \geq 0$, we can write

$$\begin{aligned}\alpha_2(\bar{K}(t)) &= \alpha_2\left(I - \frac{1}{m_t}\mathcal{L}_{\mathcal{G}(t)\times\mathcal{G}(t)}\right) = 1 - \frac{1}{m_t}\alpha_{n^2-1}(\mathcal{L}_{\mathcal{G}(t)\times\mathcal{G}(t)}) \\ &= 1 - \frac{1}{m_t}\alpha_{n^2-1}(\mathcal{L}_{\mathcal{G}(t)}) = \alpha_2\left(I - \frac{1}{m_t}\mathcal{L}_{\mathcal{G}(t)}\right) = \alpha_2(P_t), \qquad (3.18)\end{aligned}$$

where the third equality is valid because by the Perron-Frobenius theorem the smallest eigenvalue of the Laplacian of a connected graph is zero with multiplicity 1, which, in view of Lemma 2.8, shows that the second smallest eigenvalues of $\mathcal{L}_{\mathcal{G}(t)}$ and $\mathcal{L}_{\mathcal{G}(t)\times\mathcal{G}(t)}$ must be the same, i.e., $\alpha_{n^2-1}(\mathcal{L}_{\mathcal{G}(t)}) = \alpha_{n^2-1}(\mathcal{L}_{\mathcal{G}(t)\times\mathcal{G}(t)})$.

Let us now denote the largest eigenvalue and the smallest eigenvalue of a $k \times k$ matrix A by $\alpha_1(A)$ and $\alpha_k(A)$, respectively. Similarly, from Lemma 2.8 one can see that the largest eigenvalue of $\mathcal{L}_{\mathcal{G}(t)\times\mathcal{G}(t)}$ is twice that of in $\mathcal{L}_{\mathcal{G}(t)}$, i.e., $\alpha_1(\mathcal{L}_{\mathcal{G}(t)\times\mathcal{G}(t)}) = 2\alpha_1(\mathcal{L}_{\mathcal{G}(t)})$, for $t \geq 0$. Thus we can write

$$\alpha_{n^2}(\bar{K}(t)) = \alpha_{n^2}(I - \frac{1}{m_t}\mathcal{L}_{\mathcal{G}(t)\times\mathcal{G}(t)}) = 1 - \frac{1}{m_t}\alpha_1(\mathcal{L}_{\mathcal{G}(t)\times\mathcal{G}(t)}) = 1 - \frac{2}{m_t}\alpha_1(\mathcal{L}_{\mathcal{G}(t)}). \qquad (3.19)$$

In what follows, our goal is to find an upper bound for the second largest eigenvalue and a lower bound for the smallest eigenvalue of the matrix $\bar{K}(t)$.

First, we note that by relation (3.18), the second largest eigenvalue of the matrix $\bar{K}(t)$ is equal to the second largest eigenvalue of the matrix P_t. In order to bound the second largest eigenvalue of the matrix P_t we look for a relationship between its eigenvalues and the hitting times of a random walk with transition probability matrix P_t. In fact, the random target lemma (Lemma 2.2) provides us with such a relationship.

Next, in order to obtain a lower bound for the smallest eigenvalue of the matrix $\bar{K}(t)$ and in view of relation (3.19), we find an upper bound for the largest eigenvalue of the Laplacian of $\mathcal{G}(t)$, i.e., $\alpha_1(\mathcal{L}_{\mathcal{G}(t)})$. In fact, the following lemma provides us with a desired upper bound.

Lemma 3.3 *The largest eigenvalue of the Laplacian of any graph with m edges, satisfying Assumption 1, is bounded from above by $m - \frac{1}{2}$.*

Proof The proof can be found in Appendix A. □

Finally, based on Lemmas 2.2 and 3.3 we can state the main result of this section, which is an upper bound for the expected convergence time of unbiased quantized consensus over time-varying networks.

Theorem 3.5 *Let $m_{max} = \max_{t\geq 0} m_t$ and $D_{max} = \max_{t\geq 0} D_t$. Then, the expected convergence time of unbiased quantized consensus over time-varying graphs satisfying Assumption 1 is bounded from above by $\mathcal{O}\big(n^2 m_{max} D_{max} \log^2(n)\big)$.*

Proof Since, for all $t \geq 0$, the matrix P_t is doubly stochastic, $(\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n})$ is its stationary distribution. Using Lemma 2.2, we can write

$$\frac{1}{1-\alpha_2(P_t)} \leq \sum_{k=2}^{n} \frac{1}{1-\alpha_k(P_t)} = \frac{1}{n}\sum_{j\neq i} H_{\mathcal{Z}_t}(i,j) \leq \frac{1}{n}(n \times nm_t D_t) = nm_t D_t,$$

where the last inequality is due to the relation (3.15) for the maximum hitting time that we computed in the case of the fixed graph. Therefore, we get

$$\alpha_2(P_t) \leq 1 - \frac{1}{nm_t D_t}, \forall t \geq 0. \tag{3.20}$$

Moreover, using relation (3.19) and Lemma 3.3, we can write

$$\alpha_{n^2}(\bar{K}(t)) = 1 - \frac{2}{m_t}\alpha_1(\mathcal{L}_{\mathcal{G}(t)}) \geq 1 - \frac{2}{m_t}(m_t - \frac{1}{2}) = -1 + \frac{1}{m_t}, \ \forall t \geq 0. \tag{3.21}$$

Let the vector $p(t) = \big(p_{(1,1)}(t), p_{(1,2)}(t), \ldots, p_{(n,n)}(t)\big)'$ denote the probability at time t of being at different states of a random walk with transition matrix $\bar{K}(t)$. Since $\bar{K}(t)$ is a doubly stochastic matrix, $\pi := \pi(t) = (\frac{1}{n^2}, \frac{1}{n^2}, \ldots, \frac{1}{n^2}), \forall t \geq 0$ is its stationary distribution and the average is preserved throughout the dynamics. Now, we note that since $\bar{K}(t)$ is a real-valued and symmetric matrix, it has an orthogonal set of eigenvectors $\mathbf{1}, v_2, \ldots, v_{n^2}$, corresponding to the eigenvalues $1 > \alpha_2(\bar{K}(t)) \geq \cdots \geq \alpha_{n^2}(\bar{K}(t))$. Since $(p(t)-\pi)'\mathbf{1} = 0$, $p(t)-\pi$ can be written as $\sum_{k=2}^{n^2} r_k v_k$ for some coefficients $r_k, k = 2, \ldots, n^2$. In particular, since $v_k, k = 2, \ldots, n^2$, are orthogonal, we have $\|p(t)-\pi\|_2^2 = \sum_{k=2}^{n^2} r_k^2$. Now we can write

$$\begin{aligned}
\big\|\bar{K}(t)p(t)-\pi\big\|_2^2 = \big\|\bar{K}(t)(p(t)-\pi)\big\|_2^2 &= \left\|\sum_{k=2}^{n^2} r_k(\bar{K}(t)v_k)\right\|_2^2 \\
&= \left\|\sum_{k=2}^{n^2} r_k\alpha_k(\bar{K}(t))v_k\right\|_2^2 = \sum_{k=2}^{n^2} r_k^2\alpha_k^2(\bar{K}(t)) \\
&\leq \max_{k=2,\ldots,n^2}\big\{\alpha_k^2(\bar{K}(t))\big\}\sum_{k=2}^{n} r_k^2 \\
&= \max\big\{\alpha_2^2(\bar{K}(t)), \alpha_{n^2}^2(\bar{K}(t))\big\}\,\|p(t)-\pi\|_2^2 .
\end{aligned}$$

Therefore, for every probability vector $p(t)$, we have

$$\begin{aligned}\left\|\bar{K}(t)p(t)-\pi\right\|_2 &\leq \max\left\{|\alpha_2(\bar{K}(t))|, |\alpha_{n^2}(\bar{K}(t))|\right\}\|p(t)-\pi\|_2\\ &= \max\left\{|\alpha_2(P_t)|, |\alpha_{n^2}(\bar{K}(t))|\right\}\|p(t)-\pi\|_2, \qquad (3.22)\end{aligned}$$

where the equality is due to the relation (3.18). Using relations (3.20) and (3.21) in (3.22), we get

$$\begin{aligned}\left\|\bar{K}(t)p(t)-\pi\right\|_2 &\leq \max\left\{1-\frac{1}{nm_tD_t}, 1-\frac{1}{m_t}\right\}\|p(t)-\pi\|_2\\ &= \left(1-\frac{1}{nm_tD_t}\right)\|p(t)-\pi\|_2. \qquad (3.23)\end{aligned}$$

Since the above argument works for every time instant $t \geq 0$, and for each of the transition matrices P_t and $\bar{K}(t)$, using relation (3.23) recursively we get

$$\begin{aligned}\|p(t)-\pi\|_2 &= \left\|\bar{K}(t-1)\bar{K}(t-2)\dots\bar{K}(0)p(0)-\pi\right\|_2\\ &\leq \prod_{k=0}^{t-1}\left(1-\frac{1}{nm_kD_k}\right)\|p(0)-\pi\|_2\\ &\leq \left(1-\frac{1}{nm_{max}D_{max}}\right)^t\|p(0)-\pi\|_2\\ &= \left(1-\frac{1}{nm_{max}D_{max}}\right)^t\sqrt{\frac{n^2-1}{n^2}}\\ &\leq e^{\frac{-t}{nm_{max}D_{max}}}\sqrt{\frac{n^2-1}{n^2}} < e^{\frac{-t}{nm_{max}D_{max}}},\end{aligned}$$

where $p(0)$ denotes the initial probability, which is 1 in one entry and zero everywhere else. Therefore after at most $2nm_{max}D_{max}(1+\log(n))$ steps, we get $\left(\frac{1}{2}\right)^{\frac{t}{nm_{max}D_{max}}} \leq \frac{1}{2n^2}$ and this means $\|p(t)-\pi\|_2 \leq \frac{1}{2n^2}$. In other words, for all $t \geq 2nm_{max}D_{max}(1+\log(n))$, we must have $p_{(i,j)}(t) \in [\frac{1}{2n^2}, \frac{3}{2n^2}], \forall i, j$. In particular, $\sum_{i=1}^{n} p_{(i,i)}(t) \geq n \times \frac{1}{2n^2} = \frac{1}{2n}$. This means that after at most $2nm_{max}D_{max}(1+\log(n))$ steps the probability of hitting at least one of the states in $S = \{(i,i), i = 1\dots, n\}$ is larger than or equal to $\frac{1}{2n}$. Now, by applying Proposition 4.1 in [16], we conclude that the expected hitting time of a random walk with transition probabilities $\bar{K}(t)_{t\geq 0}$ is less than or equal to $4n \times 2nm_{max}D_{max}(1+\log(n))$. This result can be also viewed as a corollary of Lemma 13 in [13]. Therefore, since $\{\bar{K}(t)\}_{t\geq 0}$ and $\{K(t)\}_{t\geq 0}$ have the same expected hitting times to the states of S, we conclude that the expected meeting time of the virtual process started from any time step $t \geq 0$, which we denote by M_t^{v}, is bounded from above by $M_t^{\mathrm{v}} \leq 8n^2m_{max}D_{max}(1+\log(n))$.

Now, as in the case of static graphs, we argue that the virtual process and the original process are the same until the two walkers are each other's neighbors, i.e. for some x, y and $t \geq 0$, with $x \in N_t(y)$. At this time, the probability that two walkers in the original process meet each other at the next time step is at least half of that in the virtual process. In other words, more than half of the times when two walkers are each other's neighbors and they meet in the virtual process, they will meet in the original process as well. Since each of these intersections may happen independently, we can write

$$\begin{aligned}\overline{T} &\leq \sum_{k=0}^{\infty}(\frac{1}{2})^k \max_{t_1<t_2<\cdots<t_k}(M^{\text{v}}_{t_1}+M^{\text{v}}_{t_2}+\cdots+M^{\text{v}}_{t_k})\\ &\leq \sum_{k=0}^{\infty}(\frac{1}{2})^k k\times 8n^2 m_{max}D_{max}(1+\log(n))\\ &= 16n^2 m_{max}D_{max}(1+\log(n)). \end{aligned} \tag{3.24}$$

Finally, similar to the proof of Theorem 3.4, one can see that after an expected time of at most $O(\overline{T}\times\log(n))$, the Lyapunov function $V(t)=\max_i x_i(t)-\min_i x_i(t)$ will decrease by at least 1 through the trajectory of the dynamics over time-varying networks. This, in view of the relation (3.24) completes the proof. □

In fact, Theorem 3.5 improves significantly some of the existing upper bounds for the expected convergence time of quantized consensus over time-varying networks [16]. Finally, we would like to emphasize that as in the analysis of static networks and using direct analysis for the two special cases of star graph and double-star graph (Lemma A.1), one may be able to generalize Theorem 3.5 to such graphs.

3.4 Conclusion

In this chapter, we have studied the unbiased quantized consensus problem under the assumption that the underlying network $\mathcal{G}$ is connected. We provided tight upper and lower bounds (up to a constant factor) for the maximum expected convergence time of the model based on the effective resistances of the underlying network. We observed that the given bounds for static networks agree with the simulation results for some particular choices of undirected connected networks. Finally, we extended our results to time-varying networks under the assumption of connectivity over the sequence of networks, and provided an upper bound which significantly improves the best known bound for the randomized quantized consensus over time-varying networks.

References

1. A. Kashyap, T. Başar, and R. Srikant, "Quantized consensus," *Automatica*, vol. 43, no. 7, pp. 1192–1203, 2007.
2. J. Lorenz, *Repeated averaging and bounded-confidence, modeling, analysis and simulation of continuous opinion dynamics*. Ph.D. Dissertation, University of Bremen, 2007.
3. A. Nedic, A. Olshevsky, A. Ozdaglar, and J. N. Tsitsiklis, "On distributed averaging algorithms and quantization effects," *IEEE Transactions on Automatic Control*, vol. 54, no. 11, pp. 2506–2517, 2009.
4. R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1520–1533, 2004.
5. L. Xiao, S. Boyd, and S.-J. Kim, "Distributed average consensus with least-mean-square deviation," *Journal of Parallel and Distributed Computing*, vol. 67, no. 1, pp. 33–46, 2007.
6. M. P. Fanti, W. Ukovich, A. M. Mangini, and G. Pedroncelli, "A quantized consensus algorithm for a multi-agent assignment problem," in *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2013, pp. 1063–1068.
7. F. Bénézit, P. Thiran, and M. Vetterli, "Interval consensus: from quantized gossip to voting," in *IEEE International Conference on Acoustics, Speech and Signal Processing, 2009. ICASSP 2009*. IEEE, 2009, pp. 3661–3664.
8. B. Ghosh and S. Muthukrishnan, "Dynamic load balancing in parallel and distributed networks by random matchings," in *Proceedings of the Sixth Annual ACM Symposium on Parallel Algorithms and Architectures*. ACM, 1994, pp. 226–235.
9. D. Coppersmith, P. Tetali, and P. Winkler, "Collisions among random walks on a graph," *SIAM Journal on Discrete Mathematics*, vol. 6, pp. 363–374, 1993.
10. S. Shang, P. Cuff, P. Hui, and S. Kulkarni, "An upper bound on the convergence time for quantized consensus," in *2013 Proceedings INFOCOM*. IEEE, 2013, pp. 600–604.
11. M. Abdullah, C. Cooper, and T. Radzik, "The cover time of Cartesian product graphs," *arXiv preprint* arXiv,1202.5569, 2012.
12. R. Elsässera and T. Sauerwald, "Tight bounds for the cover time of multiple random walks," *Theoretical Computer Science*, vol. 412, pp. 2623–2641, 2011.
13. C. Avin, M. Koucký, and Z. Lotker, "How to explore a fast-changing world (cover time of a simple random walk on evolving graphs)," *Proceedings of the 35th international colloquium on Automata, Languages and Programming, Part I*, pp. 121–132, 2008.
14. D. A. Levin, Y. Peres, and E. L. Wilmer, *Markov Chains and Mixing Times*. American Mathematical Society, 2009.
15. D. Aldous and J. Fill, *Reversible Markov chains and random walks on graphs*. Book in preparation - preprint available at http://www.stat.Berkeley.EDU/users/aldous., 2002.
16. M. Zhu and S. Martinez, "On the convergence time of asynchronous distributed quantized averaging algorithms," *IEEE Transactions on Automatic Control*, vol. 56, no. 2, pp. 386–390, 2011.

Chapter 4
Metropolis Quantized Consensus

In this chapter we modify the unbiased quantized consensus algorithm introduced in Chap. 3 using Metropolis chains in order to devise a protocol with faster convergence time to the set of consensus points. Intuitively, in the unbiased quantized consensus that we studied in Chap. 3, at each time instant we choose an edge uniformly at random from the set of all the edges and let its incident nodes update their values. However, this way of selecting the edges does not take into account the specific structure of the underlying network. It turns out that if the edge selection process is done in a smarter way based on the specific structure of the graph, then one will be able to accelerate the convergence speed of the dynamics toward the consensus set. More specifically, in this chapter we show that when the edges of a static network are activated based on Poisson processes with Metropolis rates, the expected convergence time to the set of consensus points over static networks is at most $O(n^2 \log n)$, where n is the number of nodes in the network. Subsequently, we extend our results to time-varying networks and establish an upper bound of $O(n^2 \log^2 n)$ for the expected convergence time of our protocol. In turns out that these bounds are better than all previous convergence times in the literature for randomized quantized consensus [1].

4.1 Problem Formulation

In this section, we assume we are given a fixed, undirected, connected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ without self-loops. We adopt the convention of using n for the number of nodes in this graph, i.e., $n = |\mathcal{V}|$. The Metropolis Markov chain on this graph is then defined as follows.

Let $\mathcal{M}$ be a square matrix whose ij'th entry is defined as

$$\mathcal{M}_{ij} = \begin{cases} \frac{1}{\max\{d(i),d(j)\}}, & \text{if } i \neq j,\ \ j \in N(i) \\ 1 - \sum_{j \in N(i)} \mathcal{M}_{ij}, & \text{if } i = j \\ 0 & \text{otherwise.} \end{cases} \tag{4.1}$$

S.R. Etesami, *Potential-Based Analysis of Social, Communication, and Distributed Networks*, Springer Theses, DOI 10.1007/978-3-319-54289-8_4

Note that $\mathcal{M}$ is symmetric, nonnegative, and doubly stochastic. We refer to the Markov chain which transitions according to $\mathcal{M}$ as the *Metropolis chain*. Moreover, given nodes $x, y \in V$, the expected *hitting time* $H^{\mathrm{m}}(x, y)$ is the expected time until the Metropolis chain with initial position at node x reaches node y (quantities associated with the Metropolis chain will generally be denoted with an "m" superscript). For future use and convenience, we adopt the notation $\lambda_{ij} = 1/\max(d(i), d(j))$ whenever $\{i, j\} \in \mathcal{E}$ and $\lambda_{ij} = 0$ otherwise.

4.1.1 Quantized Metropolis Dynamics

We next introduce a continuous time quantized process based on Metropolis weights, whose behavior will turn out to be related to the Metropolis chain. For each link $\{i, j\} \in \mathcal{E}$ of the graph $\mathcal{G}$, we consider a Poisson process of rate λ_{ij}. Each node i begins with an integer value $x_i(0)$ which lies in the range $[l, L]$. Each time the process corresponding to an edge $\{i, j\} \in \mathcal{E}$ registers an arrival, the two nodes i, j perform the quantized consensus update from [2]:

$$x_i(t^+) = \begin{cases} x_i(t) - 1, & \text{if } x_i(t) > x_j(t) \\ x_i(t) + 1, & \text{if } x_i(t) < x_j(t) \\ x_i(t), & \text{if } x_i(t) = x_j(t), \end{cases} \tag{4.2}$$

and likewise for node j. In other words, each $x_i(t)$ is an integer-valued jump process whose jumps occur whenever an arrival occurs at any edge incident on i.

Note that the above update rule allows for the possibility that $x_i(t^+) = x_i(t)$. In this case, we will say that the update at time t was *trivial*. Furthermore, observe that if $|x_i(t) - x_j(t)| = 1$, then the update of Eq. (4.2) will cause nodes i and j to swap values, i.e., $x_i(t^+) = x_j(t)$, $x_j(t^+) = x_i(t)$. In this case, we will also say that the update was trivial. If neither of these two cases has occurred during an update, we will say that the update was *non-trivial*. Simply speaking, a non-trivial update refers to the case where the incident nodes of an activated edge have integer values which differ by at least 2. Finally, as in Chap. 3 the objective of quantized metropolis protocol is to reach to the consensus set, i.e.,

$$C = \left\{x | x_i \in \{\lfloor \bar{x}(0) \rfloor, \lfloor \bar{x}(0) \rfloor + 1\},\ \frac{\sum_{i=1}^{n} x_i}{n} = \bar{x}(0),\ i \in [n]\right\}.$$

In fact, using Metropolis chains in the structure of our devised protocol has several advantages: it allows us to implement the protocol in a totally distributed manner. Moreover, the inherited symmetry of the Metropolis chains simplifies the convergence time analysis and makes the dynamics more tractable. Furthermore, such chains benefit from having a small expected hitting time over general networks, which is a key factor for establishing a quadratic upper bound for the expected convergence time of our designed protocol. Here, we note that since $\sum_{j\in N(i)} \lambda_{ij} \leq 1$, we have that $\sum_{\{i,j\}\in\mathcal{E}} \lambda_{ij} = O(n)$, and so the Metropolis quantized consensus protocol performs $O(n)$ updates per unit time.

4.2 Expected Convergence Time over Static Networks

We now begin the analysis of the convergence time of the quantized Metropolis algorithm over a fixed graph $\mathcal{G}$. Following similar arguments as in Sect. 3.1.1, one can see that a key step in analysis of the expected convergence time of the quantized metropolis protocol is to bound the expected time until the first nontrivial update takes place. For this purpose, and similar as before, we let $\overline{T}$ be the maximum expected time such that a nontrivial update takes place over all possible configurations of integers. Note that in this maximization we exclude the configurations that are already in the consensus set, as otherwise our goal, i.e., reaching to the consensus, is achieved (no other nontrivial update can occur). Next, similar to what we have done in Chap. 3, we define original and virtual processes in order to facilitate our analysis.

Definition 4.1 Consider two random walkers moving based on whether the activated edge in the quantized Metropolis dynamics is incident to them. That is, if one of the walkers is at node x, and if the next edge to register an arrival is incident to x, i.e., if it is $\{x, y\}$ for some $y \in N(x)$, then the random walker moves from x to y. We refer to such a process as the *original* process.

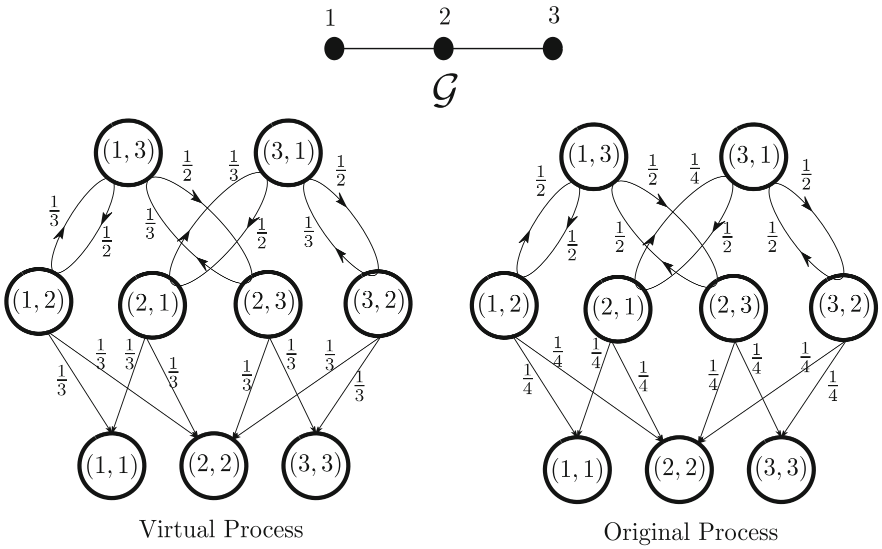
We denote by $M^{\text{o}}(x, y)$ the expected "meeting time" of this process, defined to be the expected time until an edge incident to both walkers registers an arrival provided the two walkers started at nodes x, y with $x \neq y$ (in general, we will denote quantities associated with this process with an "o" superscript). By convention, we set $M^{\text{o}}(x, x) = 0$ for all $x \in \mathcal{V}$.

Proposition 4.1 $\overline{T} = \max_{x,y\in\mathcal{V}} M^{\text{o}}(x, y)$.

Proof Proof follows exactly within the same lines of augments as in Proposition 3.1. □

Based on the above proposition, our next step is to bound $\max_{x,y\in\mathcal{V}} M^{\text{o}}(x, y)$. We will actually find it easier to instead bound a meeting time associated with a slightly different process, which we call the *virtual* process, defined next.

Definition 4.2 The virtual process is identical to the original process until the two walkers x, y become each other's neighbors in $\mathcal{G}$. At that time, the edge connecting them registers arrivals according to a Poisson process of rate $2\lambda_{xy}$ (i.e., twice the rate in the original process). The meeting time $M^{\text{v}}(x, y)$ is defined to be the expected time until an edge incident on both random walkers ticks provided the walkers start at positions x, y with $x \neq y$ (in general, we will denote quantities associated with this process with a "v" superscript). By convention we set $M^{\text{v}}(x, x) = 0$ for all $x \in \mathcal{V}$.



Virtual Process

Original Process

Fig. 4.1 Illustration of the transition probabilities in Example 4.1

Example 4.1 Let us consider a line graph with three vertices $\mathcal{V} = \{1, 2, 3\}$. We denote the position of the two walkers by a pair (x, y), where x and y refers to position of the walker 0 and walker 2, respectively. For instance, (2, 3) means that the walker 0 is at node 2 and walker 2 is at node 3. Note that the states (1, 1), (2, 2) and (3, 3) denote the situation where the walkers meet (i.e., a non-trivial update takes place). In this case the dynamics of the original and virtual processes can be completely determined by the corresponding transition probabilities of these processes given on the left and right side of the Fig. 4.1. In fact, the virtual process slightly increases the probabilities of being absorbed by the states (1, 1), (2, 2) and (3, 3). In other words, the walkers have more chance to meet in the virtual process than in the original process. However, this increase is small enough such that it keeps the virtual process as a good approximation for the original process. On the other hand, a useful fact about the virtual process is that it benefits from a more symmetric structure than the original process such that it facilitates our analysis of the hitting time and meeting time. As an example in Fig. 4.1 it can be seen that a transition from one state to its outgoing neighbors happens with equal probability for the virtual process while this property does not hold for the original process.

Definition 4.3 A vertex θ is called a hidden vertex of the Metropolis chain $\mathcal{M}$ if $H^{\mathrm{m}}(\theta, x) \leq H^{\mathrm{m}}(x, \theta), \forall x \in \mathcal{V}$.

Remark 4 It is a known fact that the hitting times of a reversible Markov chain satisfy the following transitivity property:

$$H^{\mathrm{m}}(x, y) + H^{\mathrm{m}}(y, z) + H^{\mathrm{m}}(z, x) = H^{\mathrm{m}}(x, z) + H^{\mathrm{m}}(z, y) + H^{\mathrm{m}}(y, x),$$

(see [3] for a proof). It can be shown that a consequence of this is that every reversible Markov chain has at least one hidden vertex [4].

Our first goal is to bound the expected meeting time of the virtual process, which we do in the following lemma. For this, we rely heavily on the techniques used in Chap. 3. The main idea is to argue that the expected meeting time function of the virtual process $M^{\mathrm{v}}(x, y)$ satisfies essentially the same recursion as the function $\Phi^{\mathrm{m}}(\cdot, \cdot) : \mathcal{V} \times \mathcal{V} \to \mathbb{R}$ defined by

$$\Phi^{\mathrm{m}}(x, y) = H^{\mathrm{m}}(x, y) + H^{\mathrm{m}}(y, \theta) - H^{\mathrm{m}}(\theta, y), \tag{4.3}$$

where θ denotes a fixed hidden vertex of the Metropolis chain.

Lemma 4.1 *For all $x, y \in \mathcal{V}$, we have $M^{\mathrm{v}}(x, y) \leq 6n^2$.*

Proof Fix nodes x and y such that $x \neq y$, and define

$$\Lambda_x = \sum_{x_i \in N(x)} \lambda_{xx_i}, \quad \Lambda_y = \sum_{y_i \in N(y)} \lambda_{yy_i}, \quad \Lambda_{xy} = \Lambda_x + \Lambda_y.$$

An immediate consequence of Remark 4 is the symmetry of $\Phi^m(\cdot, \cdot)$:

$$\Phi^{\mathrm{m}}(u, v) = \Phi^{\mathrm{m}}(v, u), \quad \forall\, u, v \in \mathcal{V}.$$

Therefore we can argue that

$$\begin{aligned}
\Phi^{\mathrm{m}}(x, y) &= \frac{\Lambda_x}{\Lambda_{xy}} \Phi^{\mathrm{m}}(x, y) + \frac{\Lambda_y}{\Lambda_{xy}} \Phi^{\mathrm{m}}(y, x) \\
&= \frac{\Lambda_x}{\Lambda_{xy}} \left[H^{\mathrm{m}}(x, y) + H^{\mathrm{m}}(y, \theta) - H^{\mathrm{m}}(\theta, y)\right] \\
&\quad + \frac{\Lambda_y}{\Lambda_{xy}} \left[H^{\mathrm{m}}(y, x) + H^{\mathrm{m}}(x, \theta) - H^{\mathrm{m}}(\theta, x)\right] \\
&= \frac{\Lambda_x}{\Lambda_{xy}} \left[\frac{1}{\Lambda_x} + \sum_{x_i \in N(x)} \frac{\lambda_{xx_i}}{\Lambda_x} H^{\mathrm{m}}(x_i, y)\right] + \frac{\Lambda_y}{\Lambda_{xy}} \left[\frac{1}{\Lambda_y} + \sum_{y_i \in N(y)} \frac{\lambda_{yy_i}}{\Lambda_y} H^{\mathrm{m}}(y_i, x)\right] \\
&\quad + \frac{\Lambda_x}{\Lambda_{xy}} \left[H^{\mathrm{m}}(y, \theta) - H^{\mathrm{m}}(\theta, y)\right] + \frac{\Lambda_y}{\Lambda_{xy}} \left[H^{\mathrm{m}}(x, \theta) - H^{\mathrm{m}}(\theta, x)\right],
\end{aligned}$$

where in the second equality we expanded $H^{\mathrm{m}}(x, y)$ and $H^{\mathrm{m}}(y, x)$. Simplifying the last relation, we obtain

$$\begin{aligned}
\Phi^{\mathrm{m}}(x, y) = \frac{2}{\Lambda_{xy}} &+ \sum_{x_i \in N(x)} \frac{\lambda_{xx_i}}{\Lambda_{xy}} H^{\mathrm{m}}(x_i, y) + \sum_{y_i \in N(y)} \frac{\lambda_{yy_i}}{\Lambda_{xy}} H^{\mathrm{m}}(y_i, x) \\
&+ \frac{\Lambda_x}{\Lambda_{xy}} \big(H^{\mathrm{m}}(y, \theta) - H^{\mathrm{m}}(\theta, y)\big) + \frac{\Lambda_y}{\Lambda_{xy}} \big(H^{\mathrm{m}}(x, \theta) - H^{\mathrm{m}}(\theta, x)\big).
\end{aligned} \tag{4.4}$$

Now, using the definition of $\Phi^{\mathrm{m}}(\cdot,\cdot)$, we have that for each neighbor $x_i \in N(x)$,

$$\Phi^{\mathrm{m}}(x_i, y) = H^{\mathrm{m}}(x_i, y) + H^{\mathrm{m}}(y, \theta) - H^{\mathrm{m}}(\theta, y).$$

Multiplying this relation by $\frac{\lambda_{xx_i}}{\Lambda_{xy}}$ and summing over $x_i \in N(x)$, we obtain

$$\sum_{x_i \in N(x)} \frac{\lambda_{xx_i}}{\Lambda_{xy}} \Phi^{\mathrm{m}}(x_i, y) = \sum_{x_i \in N(x)} \frac{\lambda_{xx_i}}{\Lambda_{xy}} H^{\mathrm{m}}(x_i, y) + \frac{\Lambda_x}{\Lambda_{xy}} \big[H^{\mathrm{m}}(y, \theta) - H^{\mathrm{m}}(\theta, y)\big]. \tag{4.5}$$

By the same argument, since for any $y_i \in N(y)$,

$$\Phi^{\mathrm{m}}(x, y_i) = \Phi^{\mathrm{m}}(y_i, x) = H^{\mathrm{m}}(y_i, x) + H^{\mathrm{m}}(x, \theta) - H^{\mathrm{m}}(\theta, x),$$

we have

$$\sum_{y_i \in N(y)} \frac{\lambda_{yy_i}}{\Lambda_{xy}} \Phi^{\mathrm{m}}(x, y_i) = \sum_{y_i \in N(y)} \frac{\lambda_{yy_i}}{\Lambda_{xy}} H^{\mathrm{m}}(y_i, x) + \frac{\Lambda_y}{\Lambda_{xy}} \big[H^{\mathrm{m}}(x, \theta) - H^{\mathrm{m}}(\theta, x)\big]. \tag{4.6}$$

Substituting (4.6) and (4.5) in (4.4) gives

$$\Phi^{\mathrm{m}}(x, y) = \frac{2}{\Lambda_{xy}} + \sum_{x_i \in N(x)} \frac{\lambda_{xx_i}}{\Lambda_{xy}} \Phi^{\mathrm{m}}(x_i, y) + \sum_{y_i \in N(y)} \frac{\lambda_{yy_i}}{\Lambda_{xy}} \Phi^{\mathrm{m}}(x, y_i). \tag{4.7}$$

On the other hand, we note that the meeting time of the virtual process can be expanded as

$$M^{\mathrm{v}}(x, y) = \frac{1}{\Lambda_{xy}} + \sum_{x_i \in N(x)} \frac{\lambda_{xx_i}}{\Lambda_{xy}} M^{\mathrm{v}}(x_i, y) + \sum_{y_i \in N(y)} \frac{\lambda_{yy_i}}{\Lambda_{xy}} M^{\mathrm{v}}(x, y_i). \tag{4.8}$$

Note that the above equation holds regardless of whether x and y are neighbors, and this fact is the very reason for our introduction of the virtual process.

We therefore see that $\frac{1}{2}\Phi^{\mathrm{m}}(x, y)$ and $M^{\mathrm{v}}(x, y)$ follow the same recursion formula when $x \neq y$. Thus defining $f(x, y) = \frac{1}{2}\Phi^{\mathrm{m}}(x, y) - M^{\mathrm{v}}(x, y)$ we have that for all $x, y \in \mathcal{V}, x \neq y$

$$f(x, y) = \sum_{x_i \in N(x)} \frac{\lambda_{xx_i}}{\Lambda_{xy}} f(x_i, y) + \sum_{y_i \in N(y)} \frac{\lambda_{yy_i}}{\Lambda_{xy}} f(x, y_i).$$

Defining $\mathcal{V}' = \mathcal{V} \times \mathcal{V}$ and $\mathcal{V}'' = \mathcal{V}' \setminus \{(x, x) \mid x \in \mathcal{V}\}$, we have that the matrix $\mathcal{Q}$ defined as

$$\mathcal{Q}\big((x,y),(r,s)\big) = \begin{cases} \frac{\lambda_{xr}}{\Lambda_{xy}}, & \text{if } s = y,\ r \in N(x) \\ \frac{\lambda_{yw}}{\Lambda_{xy}}, & \text{if } r = x,\ w \in N(y) \\ 0, & \text{Else} \end{cases}$$

is a stochastic irreducible matrix in $\mathbb{R}^{\mathcal{V}' \times \mathcal{V}'}$. Furthermore, we have that $f(x,y)$ is harmonic over $\mathcal{V}''$ and, since θ is a hidden vertex, we have that for all $x \in \mathcal{V}$, $f(x,x) = \frac{1}{2}(H^{\mathrm{m}}(x,\theta) - H^{\mathrm{m}}(\theta,x)) \geq 0$. Therefore, using Remark 1, we immediately get $f(x,y) \geq 0$ for all $x, y \in \mathcal{V}$, and therefore $M^{\mathrm{v}}(x,y) \leq \frac{1}{2}\Phi^{\mathrm{m}}(x,y)$. Finally, it was shown in [5] that $H^m(x,y) \leq 6n^2$ for all $x, y \in \mathcal{V}$, which now immediately implies the current lemma. □

Our next step is to argue that the bound we have just derived on $\max_{x,y} M^{\mathrm{v}}(x,y)$ in the previous lemma holds for $\max_{x,y} M^{\mathrm{o}}(x,y)$ with an additional multiplicative factor of 2. The main idea is that, by definition, the virtual process and the original process are identical until the two random walkers are each other's neighbors, and that when that happens, the probability of meeting at the next transition is $\frac{\lambda_{xy}}{\Lambda_{xy}-\lambda_{xy}}$ in the original process and $\frac{2\lambda_{xy}}{\Lambda_{xy}}$ in the virtual process. Since the former is at least half of the latter, this shows that at least half of the times when the walkers in the virtual process meet, they will meet in the original process as well. This immediately implies that the meeting time of the original process lies within a constant factor of that in the virtual process which by Lemma 4.1 is at most $O(n^2)$. To provide a formal proof, we state the following lemma.

Lemma 4.2 $\overline{T} := \max_{x,y} M^{\mathrm{o}}(x,y) \leq 12n^2$.

Proof Fix x, y with $x \neq y$ and initialize both an original process and a virtual process with initial positions of the walkers being x and y. Furthermore, let us couple these processes to move identically until the first time when the walkers are each other's neighbors; clearly, this has no effect on the distribution of either process. When the walkers are each other's neighbors for the first time, say at nodes u and w, let us split the edge connecting them of rate $2\lambda_{uw}$ in the virtual process into two distinct edges of rate λ_{uw} (Fig. 4.2). We will make the first of these (the bottom black edge in the right side of Fig. 4.2), as well as the edges going from u and w to their neighbors, register arrivals in the virtual process exactly when the corresponding edges in the original process register arrivals. Again, this has no effect on the distribution of either process.

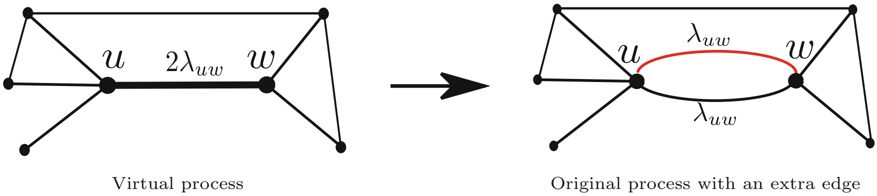


Fig. 4.2 Converting the virtual process to the original process plus an extra edge

Now, for these coupled processes, let us denote by $\tau^{\mathrm{v}}(x, y)$ a random variable denoting the first time when the random walkers in the virtual processes meet (recall this means that an edge incident on both of them registers an arrival) and $\tau^{\mathrm{o}}(x, y)$ the first time when the random walkers in the original process meet, given that the processes are initialized from x and y. We have that $\tau^{o}(x, y) = \tau^{\mathrm{v}}(x, y)$ if the first edge of rate λ_{uw} registers an arrival before the second edge of rate λ_{uw}. This happens with probability 1/2. If this does not happen, $\tau^{\mathrm{o}}(x, y)$ is $\tau^{\mathrm{v}}(x, y)$ plus another random variable which is a meeting time in the original process starting from u and w, i.e., $\tau^{\mathrm{o}}(u, w)$. In other words,

$$\tau^{\mathrm{o}}(x, y) = \begin{cases} \tau^{\mathrm{v}}(x, y), & \text{w.p. } \frac{1}{2}(\text{black edge ticks}) \\ \tau^{\mathrm{v}}(x, y) + \tau^{\mathrm{o}}(u, w) & \text{w.p. } \frac{1}{2}(\text{red edge ticks}). \end{cases} \tag{4.9}$$

Thus, by taking expectation from both sides of relation (4.9) we get

$$M^{\mathrm{o}}(x, y) \leq \frac{1}{2} M^{\mathrm{v}}(x, y) + \frac{1}{2}(M^{\mathrm{v}}(x, y) + \max_{u,w} M^{\mathrm{o}}(u, w))$$

which implies that $\max_{x,y} M^{\mathrm{o}}(x, y) \leq 2 \max_{x,y} M^{\mathrm{v}}(x, y) \leq 12n^2$, where the last step made use of Lemma 4.1. □

Remark 5 Let $\tau^{\mathrm{o}}(x, y)$ be the meeting time of two random walkers in the original process, who start at x and y. In view of the Lemma 2.3, we immediately have that

$$\mathbb{P}(\tau^{\mathrm{o}}(x, y) > t) \leq e^{-\lfloor \frac{t}{12en^2} \rfloor}. \tag{4.10}$$

Indeed, this follows by dividing the time interval of length t into $12en^2$ intervals and applying Markov's inequality in each one. For a rigorous justification, see Chap. 2.4.3 of [3] whose argument applies verbatim here.

Now we are in a position to state the main result of this section, which is an upper bound for the expected convergence time of the quantized Metropolis protocol.

Theorem 4.2 *The expected time until the quantized Metropolis dynamics reach a consensus set is* $O(n^2 \log n)$.

Proof The proof is quite similar to that in Theorem 3.4. Basically, the idea is to consider the time it takes for the Lyapunov function $V(t) = \max_i x_i(t) - \min_i x_i(t)$ to shrink by at least 1 starting from any non-consensus configuration, and showing that this time is $O(n^2 \log n)$, which will prove the statement of the theorem. □

4.3 Expected Convergence Time over Time-Varying Networks

In this section we analyze the expected convergence time of the quantized Metropolis dynamics over time-varying networks. Toward this aim, let us consider a sequence of connected undirected networks $\mathcal{G}(t) = (\mathcal{V}, \mathcal{E}(t)), t \geq 0$, over the same set of vertices V which may change at discrete time instances, i.e., $\mathcal{G}(t) = \mathcal{G}(\lfloor t \rfloor), \forall t \geq 0$. Thus our notation for the set of neighbors of x will now be $N_x(t)$, which depends on t. In contrast to the previous section, we will now assume that each node always possesses a self-loop, i.e., $x \in N_x(t)$ for all $x \in \mathcal{V}, t \geq 0$. We thus introduce the notation $N'_x(t)$ to denote the neighbors of node x excluding itself: $N'_x(t) = N_x(t) \setminus \{x\}$. The degree of a node x, which we will denote by $d_x(t)$, will refer to the cardinality of $N'_x(t)$.

4.3.1 *Quantized Metropolis Model over Time-Varying Networks*

Given a network $\mathcal{G}(t)$ at time $t \geq 0$, we associate a Poisson process with each edge $\{i, j\}$ of $\mathcal{G}(t)$ with a rate of $\lambda_{ij}(t) = 1/\max(d_i(t), d_j(t))$, i.e., the Metropolis weight corresponding to that edge at time t. When an edge $\{i, j\} \in \mathcal{E}(t)$ registers an arrival, we let the incident nodes update their values based on (4.2). Moreover, for each node x and time t, the self-loop (x, x) registers arrivals according to a Poisson process with rate of

$$\lambda_{xx}(t) = 1 - \frac{1}{2} \sum_{x_i \in N'_x(t)} \lambda_{xx_i}(t). \tag{4.11}$$

Note that $\lambda_{xx}(t) \geq 0$ for all $x \in \mathcal{V}, t \geq 0$, and the sum of the rates of the edges of $\mathcal{G}(t)$ is always equal to $|V| = n$. When a self-loop registers an arrival, no update is made.

4.3.2 *Preliminary Definitions and Relevant Results*

We next introduce the notions of the “original” and “virtual” processes over the time-varying graph sequence $\{\mathcal{G}(t)\}$.

Definition 4.4 Consider two random walkers moving based on whether the activated edge in the quantized Metropolis location is incident to them or not. That is, if a random walker is at node i at time t when the edge $\{i, j\} \in \mathcal{G}(t)$ registers an arrival, the walker moves to node j. As before, we will refer to this as the original process, but note that the graph sequence is now time-varying. The meeting time $M_t^{\text{o}}(x, y)$

for $x \neq y$ is the expected time until two random walkers, starting at locations x and y at time t, are both incident on an edge which registers an arrival. By convention we set $M_t^{\mathrm{o}}(x, x) = 0$ for all $x \in \mathcal{V}, t \geq 0$.

Definition 4.5 We define the virtual process to be identical to the original process except when the two walkers in the original process are each other's neighbors at nodes u and w for some $t \geq 0$, i.e., $u \in N_w(t)$. At this time in the virtual process we let the edges $\{r, s\} \in \mathcal{E}(t)$ register arrivals at rates $\mu_{rs}(t)$ defined as follows:

$$\mu_{rs}(t) = \begin{cases} 2\lambda_{uw}(t), & \text{if } \{r, s\} = \{u, w\} \\ \lambda_{uu}(t) - \frac{\lambda_{uw}(t)}{2}, & \text{if } \{r, s\} = \{u, u\} \\ \lambda_{ww}(t) - \frac{\lambda_{uw}(t)}{2}, & \text{if } \{r, s\} = \{w, w\} \\ \lambda_{rs}(t), & \text{if else.} \end{cases} \tag{4.12}$$

Note that $\mu_{rs}(t) \geq 0$ for all $r, s \in \mathcal{V}, t \geq 0$. We refer to the meeting time of the virtual process starting from x and y at time t as $\tau_t^{\mathrm{v}}(x, y)$ and its expectation $M_t^{\mathrm{v}}(x, y)$. By convention, we set $M_t^{\mathrm{v}}(x, x) = 0$ for all $x \in \mathcal{V}, t \geq 0$.

In words, the rate of the edge $\{u, w\}$ is doubled, while the rate of the self-loops is decreased to make sure that all rates still sum up to n.

4.3.3 Convergence Rate over Time-Varying Networks

In this part, we state our main results for the quantized Metropolis dynamics over time-varying networks. Define $\Lambda_{xy}(t)$ to be

$$\Lambda_{xy}(t) = \sum_{x_i \in N_x'(t)} \lambda_{xx_i}(t) + \sum_{y_i \in N_y'(t)} \lambda_{yy_i}(t) = 2\left(2 - \lambda_{xx}(t) - \lambda_{yy}(t)\right). \tag{4.13}$$

Fix t and let T_1 be the first time after t that an edge registers an arrival in the virtual process. We then have

$$\begin{aligned} \mathbb{E}\left[\tau_t^{\mathrm{v}}(x, y) | T_1\right] &= (T_1 - t) + \left(1 - \frac{\Lambda_{xy}(T_1)}{n}\right) M_{T_1}^{\mathrm{v}}(x, y) \\ &+ \sum_{x_i \in N_x'(T_1)} \frac{\lambda_{xx_i}(T_1)}{n} M_{T_1}^{\mathrm{v}}(x_i, y) + \sum_{y_i \in N_y'(T_1)} \frac{\lambda_{yy_i}(T_1)}{n} M_{T_1}^{\mathrm{v}}(x, y_i). \end{aligned} \tag{4.14}$$

Note that this equation holds regardless of whether x and y are neighbors, and this fact is why we introduced the virtual process in the first place. Next in the following lemma we show that T_1 follows an exponential distribution with parameter n.

Lemma 4.3 *Let us consider a process obtained by restarting a Poisson process of rate n at every discrete time instant $k = 0, 1, 2, \ldots$. The first time arrival T_1 of the new process thus generated will follow an exponential distribution with parameter n.*

Proof We show that $\mathbb{P}(T_1 \leq t) = 1 - e^{-nt}, \forall t \geq 0$. For $0 \leq t < 1$ the claim is trivial since the first time arrival of a single Poisson process follows an exponential distribution. Now given that for $t \in [0, k)$ we have $\mathbb{P}(T_1 \leq t) = 1 - e^{-nt}$, we will compute $\mathbb{P}(T_1 \leq t)$ for $t \in [k, k + 1)$. In fact, for $t \in [k, k + 1)$ we can write

$$\mathbb{P}(T_1 \leq t) = \mathbb{P}(T_1 \leq k) + \mathbb{P}(k \leq T_1 \leq t) = (1 - e^{-nk}) + \mathbb{P}(k \leq T_1 \leq t). \quad (4.15)$$

Since $k \leq T_1 \leq t$ if and only if there is no arrival in any of the past k intervals $[0, 1), [1, 2), \ldots, [k - 1, k)$ and the first arrival must occur in $[k, t) \subseteq [k, k + 1)$, and since these events are independent due to restarting of the Poisson process at the beginning of each interval, we can write

$$\begin{aligned}\mathbb{P}(k \leq T_1 \leq t) &= \prod_{i=0}^{k-1} \mathbb{P}(T_1 \notin [i, i + 1)) \times \mathbb{P}(k \leq T_1 < t)\\ &= (e^{-n})^k \times (1 - e^{-n(t-k)})\\ &= e^{-nk} + e^{-nt}. \end{aligned} \quad (4.16)$$

By substituting (4.16) into (4.15) we get $\mathbb{P}(T_1 \leq t) = 1 - e^{-nt}$. This completes the proof. □

Continuing with (4.14) and since T_1 is exponential with parameter n, we further have that

$$\begin{aligned} M_t^{\mathrm{v}}(x, y) &= \int_t^{\infty} n e^{-n(t_1 - t)} \mathbb{E}\left[\tau_t^{\mathrm{v}}(x, y) | T_1 = t_1\right] dt_1\\ &= \frac{1}{n} + \int_t^{\infty} n e^{-n(t_1 - t)} \left(1 - \frac{\Lambda_{xy}(t_1)}{n}\right) M_{t_1}^{\mathrm{v}}(x, y) dt_1\\ &\quad + \int_t^{\infty} \sum_{x_i \in N_x'(t_1)} n e^{-n(t_1 - t)} \frac{\lambda_{x x_i}(t_1)}{n} M_{t_1}^{\mathrm{v}}(x_i, y) dt_1\\ &\quad + \int_t^{\infty} \sum_{y_i \in N_y'(t_1)} n e^{-n(t_1 - t)} \frac{\lambda_{y y_i}(t_1)}{n} M_{t_1}^{\mathrm{v}}(x, y_i) dt_1. \end{aligned} \quad (4.17)$$

Now let us define $v(t)$ to be a column vector of length $n(n - 1)$ whose entries are the variables $M_t^{\mathrm{v}}(x, y), \forall x \neq y$ in any order. It follows that we can write the above recursion for $M_t^{\mathrm{v}}(x, y)$ in matrix form as

$$v(t) = \frac{1}{n}\mathbf{1} + \int_t^{\infty} n e^{-n(t_1 - t)} D\,(t_1)\, v\,(t_1)\; dt_1, \quad (4.18)$$

where $D(t) \in \mathbb{R}^{n(n-1)\times n(n-1)}$ is a matrix whose rows and columns we will index by $(x, y), x \neq y$ as

$$D_{(x,y)(r,s)}(t) = \begin{cases} \frac{\lambda_{rx}(t)}{n}, & \text{if } s = y, r \in N_x(t) \setminus \{x, y\} \\ \frac{\lambda_{sy}(t)}{n}, & \text{if } r = x, s \in N_y(t) \setminus \{x, y\} \\ 1 - \frac{\Lambda_{xy}(t)}{n}, & \text{if } (r, s) = (x, y) \\ 0, & \text{Else.} \end{cases} \tag{4.19}$$

We change variables in (4.18) from t_1 to $z = -e^{-n(t_1 - t)}$ and obtain

$$v(t) = \frac{1}{n}\mathbf{1} + \int_{-1}^{0} D\left(t - \frac{\ln(-z)}{n}\right) v\left(t - \frac{\ln(-z)}{n}\right) dz. \tag{4.20}$$

We justify this change of variables by appealing to Theorem 263I in [6]. Indeed, the equivalence of Eqs. (4.20) and (4.18) is an instance of the equality $\int_{I'} g = \int_I g(\phi(z))\phi'(z)$. Here $\phi(z) = -e^{-n(z-t)}$ while $g(z) = D(t - \ln(-z)/n)v(t - \ln(-z)/n)$. Theorem 263I in [6] allows us to assert this equality subject to (i) $g(\cdot)$ being Lebesgue measurable (ii); and $\phi(z)$ being absolutely continuous on any closed bounded subinterval of I. Item (ii) is clearly satisfied here. Item (i) follows because $D(\cdot)$ is a piecewise continuous function by definition, and $v(\cdot)$ is a continuous function (indeed, taking $t_n \to t$ and conditioning on no transitions in $[t_n, t]$ as well as no meeting occurring in the same interval, we immediately obtain the continuity of $v(t)$ as a function of t).

Next, we note that it is immediate from Eq. (4.19) that $D(t)$ is a symmetric matrix. Furthermore, it is easy to see that $D(t)$ is sub-stochastic,[1] implying that its eigenvalues are all real and less than 1 in modulus. Our next step is to upper bound both the largest and smallest eigenvalues of $D(t)$.

The first step is to extend $D(t)$ to a stochastic matrix $P \in \mathbb{R}^{n^2 \times n^2}$ as follows:

$$P_{(x,y)(r,s)}(t) = \begin{cases} 1, & \text{if } x = y = r = s \\ \frac{\lambda_{rx}(t)}{n}, & \text{if } x \neq y, s = y, r \in N_x(t) \setminus \{x\} \\ \frac{\lambda_{sy}(t)}{n}, & \text{if } x \neq y, r = x, s \in N_y(t) \setminus \{y\} \\ 1 - \frac{\Lambda_{xy}(t)}{n}, & \text{if } x \neq y, (r, s) = (x, y), \\ 0, & \text{if else.} \end{cases} \tag{4.21}$$

Note that since $P(t)$ is stochastic, it can be interpreted as the transition matrix of a Markov chain with absorbing states $S = \{(x, x) : x \in \mathcal{V}\}$. When we adopt the convention that states in S correspond to the first n rows of $P(t)$, we have that the matrix $P(t)$ can be represented as

[1] A matrix with nonnegative entries is sub-stochastic if each row sum to at most 1.

$$P(t) = \begin{pmatrix} I_{n\times n} & 0 \\ C(t) & D(t) \end{pmatrix}, \tag{4.22}$$

where $C(t)$ and $D(t)$ are, respectively, matrices of sizes $(n^2 - n) \times n$ and $(n^2 - n) \times (n^2 - n)$.

Example 4.2 Let us consider a line graph of three nodes, $\mathcal{V} = \{1, 2, 3\}$, and the set of edges $\mathcal{E} = \{\{1, 2\}, \{2, 3\}, \{1, 1\}, \{2, 2\}, \{3, 3\}\}$. In this case we have $\lambda_{22} = \lambda_{12} = \lambda_{23} = \frac{1}{2}$, and $\lambda_{11} = \lambda_{33} = \frac{3}{4}$. If the two random walkers are not each other's neighbors, these are the rates at which edges register arrivals. However, if the walkers are each other's neighbors, then modifications must be made as we detailed above. For example, suppose $x = 1$ and $y = 2$, then we have $\mu_{12} = 1$, $\mu_{11} = \mu_{23} = \frac{1}{2}$, $\mu_{22} = \frac{1}{4}$, and $\mu_{33} = \frac{3}{4}$ (Fig. 4.3). In particular, the transition probabilities in matrix P for such a network is as follows:

$$\begin{array}{c|ccc|cccccc}
 & (1,1) & (2,2) & (3,3) & (1,2) & (1,3) & (2,1) & (2,3) & (3,1) & (3,2) \\
(1,1) & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
(2,2) & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
(3,3) & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
(1,2) & \frac{1}{6} & \frac{1}{6} & 0 & \frac{1}{2} & \frac{1}{6} & 0 & 0 & 0 & 0 \\
(1,3) & 0 & 0 & 0 & \frac{1}{6} & \frac{2}{3} & 0 & \frac{1}{6} & 0 & 0 \\
(2,1) & \frac{1}{6} & \frac{1}{6} & 0 & 0 & 0 & \frac{1}{2} & 0 & \frac{1}{6} & 0 \\
(2,3) & 0 & \frac{1}{6} & \frac{1}{6} & 0 & \frac{1}{6} & 0 & \frac{1}{2} & 0 & 0 \\
(3,1) & 0 & 0 & 0 & 0 & 0 & \frac{1}{6} & 0 & \frac{2}{3} & \frac{1}{6} \\
(3,2) & 0 & \frac{1}{6} & \frac{1}{6} & 0 & 0 & 0 & 0 & \frac{1}{6} & \frac{1}{2}
\end{array},$$

where the left-down and right-down blocks represent, respectively, matrices D and C in the block form representation given in (4.22). As an example, to find $P_{((1,2),(1,3))}$, one can either simply use (4.21) to obtain $P_{((1,2),(1,3))} = \frac{\lambda_{23}}{3} = \frac{1}{6}$, or to compute the probability of transition of the virtual process in the right-bottom of Fig. 4.3 from state $(1, 2)$ to $(1, 3)$, which is equal to $P_{((1,2),(1,3))} = \frac{\mu_{23}}{3} = \frac{1}{6}$. Similarly, to find $P_{((1,2),(1,2))}$ we can write $P_{((1,2),(1,2))} = 1 - \frac{2\lambda_{12}+\lambda_{23}}{3} = \frac{1}{2}$, which is equal to the probability that the virtual process in the right-bottom of Fig. 4.3 does not move, i.e., $P_{((1,2),(1,2))} = \frac{\mu_{11}+\mu_{22}+\mu_{33}}{3} = \frac{\frac{1}{2}+\frac{1}{4}+\frac{3}{4}}{3} = \frac{1}{2}$.

Lemma 4.4 *Consider a Markov chain with transition matrix* $Q = \begin{pmatrix} I_{k\times k} & 0 \\ C & D \end{pmatrix}$, *which has the additional property that there is a path starting from any node and ending in the first k nodes. Furthermore, let us assume that D is symmetric and let us denote its largest eigenvalue by* $\lambda_{\max}(D)$. *Let* H_i *denote the expected time until the chain is absorbed in* $\{1, \ldots, k\}$ *starting at node i and* $H = \max_i H_i$. *Then* $\lambda_{\max}(D) \leq 1 - \frac{1}{H}$.

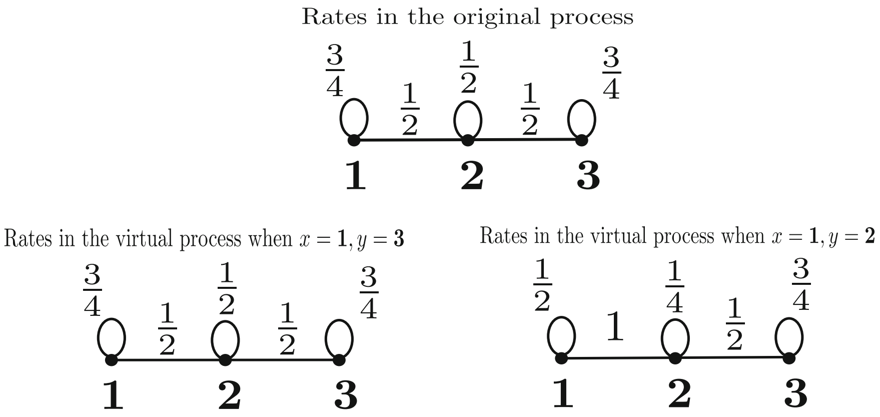

Rates in the original process
3/4
1/2
1/2
1/2
3/4
1
2
3
Rates in the virtual process when $x = 1, y = 3$
3/4
1/2
1/2
1/2
3/4
1
2
3
Rates in the virtual process when $x = 1, y = 2$
1/2
1
1/4
1/2
3/4
1
2
3

Fig. 4.3 Rates of the original and virtual processes in Example 4.2

Proof Indeed, $1/(1 - \lambda_{\max}(D))$ is a positive number which is an eigenvalue of $(I - D)^{-1}$ and is consequently upper bounded by $||(I - D)^{-1}||_\infty$. But by [7, Theorem 4.5], we have that the sum of the entries in the ith row of $(I - D)^{-1}$ is H_i. □

Lemma 4.5 *For all $t \geq 0$, we have $\lambda_{\max}(D(t)) \leq 1 - \frac{1}{6n^3}$.*

Proof Fix t and let $H^{\mathrm{P(t)}}(\{x, y\})$ denote the hitting time to $S = \{\{i, i\} \mid i \in V\}$ in $P(t)$ starting from $\{x, y\}$. It is immediately apparent that when $x \neq y$,

$$H^{\mathrm{P(t)}}(\{x, y\}) = \frac{n}{\Lambda_{xy}(t)} + \sum_{x_i \in N'_x(t)} \frac{\lambda_{xx_i}(t)}{\Lambda_{xy}(t)} H^{\mathrm{P(t)}}(\{x_i, y\}) + \sum_{y_i \in N'_y(t)} \frac{\lambda_{yy_i}(t)}{\Lambda_{xy}(t)} H^{\mathrm{P(t)}}(\{x, y_i\}).$$

Comparing this to Eq. (4.8) and noting that $H^{P(t)}(\{i, i\}) = 0$ for all $i \in V$, we obtain that $H^{P(t)}(\{x, y\})/n$ equals the expected meeting time in the virtual process on the graph $\mathcal{G}(t)$ starting from x and y. Appealing to Lemma 4.1, we can conclude that expected hitting time in $P(t)$ to the first n states is at most $6n^3$. Appealing to Lemma 4.4 then completes the proof. □

Observe that the lower bound

$$\lambda_{\min}(D(t)) \geq 1 - \frac{4}{n} \tag{4.23}$$

follows immediately by Gershgorin circles theorem [8] due to the observation that $\Lambda_{xy}(t) \leq 2$ for all $x, y \in \mathcal{V}$, $t \geq 0$.

We are now in a position to state and prove the main result of this section.

Theorem 4.3 *The expected time until the quantized Metropolis dynamics over time-varying connected networks reach the consensus set is $O(n^2 \log^2(n))$.*

Proof Note that we may iterate the recursion of Eq. (4.20) to obtain

$$\begin{aligned} v(t) &= \frac{1}{n}\mathbf{1} + \int_{-1}^{0} D\left(t - \frac{\ln(-z)}{n}\right)\left(\frac{1}{n}\mathbf{1} + \int_{-1}^{0} D\left(t - \frac{\ln(-z)}{n} - \frac{\ln(-z')}{n}\right)\right. \\ &\quad \left. \times\, v\left(t - \frac{\ln(-z)}{n} - \frac{\ln(-z')}{n}\right) dz'\right) dz \\ &\le \frac{2}{n}\mathbf{1} + \int_{-1}^{0}\int_{-1}^{0} D\left(t - \frac{\ln(-z)}{n}\right) D\left(t - \frac{\ln(-z)}{n} - \frac{\ln(-z')}{n}\right) \\ &\quad \times\, v\left(t - \frac{\ln(-z)}{n} - \frac{\ln(-z')}{n}\right)\, dz'dz, \end{aligned}$$

where the last step used the sub-stochasticity of $D(\cdot)$. Iterating this k times, we obtain

$$v(t) \le \frac{k}{n}\mathbf{1} + \int_{[-1,0]^k} \left[\prod_{j=1}^{k} D\left(t - \sum_{i=1}^{j} \frac{\ln(-z_i)}{n}\right)\right] \cdot v\left(t - \sum_{i=1}^{k}\frac{\ln(-z_i)}{n}\right) dz_k \ldots dz_1. \tag{4.24}$$

Now let us use introduce the notation $M = \sup_{t\ge 0} \|v(t)\|_\infty$. Elementary arguments suffice to establish that $M < \infty$; recall that $v(t)$ stacks up expected meeting times in the virtual process starting at time t, and the finiteness of M can follow from the observation that the probability of intersection in any positive length interval $[t, t+a]$ is positive and bounded away from zero independently of t. Thus for every $\epsilon > 0$ there exists t^* such that $M - \epsilon \le \|v(t^*)\|_\infty$. Since the entries of $D(\cdot)$ and $v(\cdot)$ are always nonnegative, we can take the infinity-norm from both sides of (4.24) and use triangle inequality to obtain

$$\begin{aligned} M - \epsilon \le \|v(t^*)\|_\infty &\le \frac{k}{n} + \left\|\prod_{j=1}^{k} D\left(t^* - \sum_{i=1}^{j}\frac{\ln(-z_i)}{n}\right)\right\|_\infty \left\|v\left(t^* - \sum_{i=1}^{k}\frac{\ln(-z_i)}{n}\right)\right\|_\infty dz_k \ldots dz_1 \\ &\le \frac{k}{n} + M \iint\ldots\int \left\|\prod_{j=1}^{k} D\left(t^* - \sum_{i=1}^{j}\frac{\ln(-z_i)}{n}\right)\right\|_\infty dz_k \ldots dz_1 \\ &\le \frac{k}{n} + M \iint\ldots\int \sqrt{n(n-1)} \left\|\prod_{j=1}^{k} D\left(t^* - \sum_{i=1}^{j}\frac{\ln(-z_i)}{n}\right)\right\|_2 dz_k \ldots dz_1 \\ &\le \frac{k}{n} + M \iint\ldots\int \sqrt{n(n-1)} \prod_{j=1}^{k} \left\|D\left(t^* - \sum_{i=1}^{j}\frac{\ln(-z_i)}{n}\right)\right\|_2 dz_k \ldots dz_1 \\ &\le \frac{k}{n} + M\sqrt{n(n-1)}\left(\sup_{t\ge 0}\{|\lambda_{\max}(D(t))|, |\lambda_{\min}(D(t))|\}\right)^k \\ &\le \frac{k}{n} + M\sqrt{n(n-1)}\left(1 - \frac{1}{6n^3}\right)^k, \end{aligned} \tag{4.25}$$

where in the fourth inequality we have used the fact that the infinity-norm of a matrix is always upper bounded by its induced 2-norm times the square root of its dimension, i.e., $\|D(\cdot)\|_\infty \leq \sqrt{n(n-1)}\|D(\cdot)\|_2$. Moreover, the last inequality is valid due to Lemma 4.5 and Eq. (4.23).

Let us choose k^* so that

$$\sqrt{n(n-1)}\left(1 - \frac{1}{6n^3}\right)^{k^*} \leq \frac{1}{2}.$$

Appealing to the inequality $(1 - 1/x)^x \leq e^{-1}$, we obtain that we may choose $k^* = O(n^3 \log n)$ to accomplish this. Plugging this into the last line of Eq. (4.25), we immediately obtain

$$M - \epsilon \leq \frac{k^*}{n} + \frac{M}{2}.$$

Since this holds for any $\epsilon > 0$, it implies that $M \leq 2k^*/n = O(n^2 \log n)$.

We have thus obtained an upper bound on $O(n^2 \log n)$ on the expected meeting time in the virtual process starting from any pair of nonidentical nodes and any time t. The rest of the proof directly parallels the arguments we have made in the fixed graph case, and we only sketch it rather than repeat the relevant sections verbatim. The first step is to argue that $2M$ is an upper bound on the meeting time of the original process. The proof proceeds exactly as in the case of Lemma 4.2 by coupling the two processes and conditioning on the transition at the last step in the meeting time of the virtual process. The next (and final) step is to argue that the time it takes $V(t) = \max_i x_i(t) - \min_i x_i(t)$ to shrink by 1 is upper bounded by $O(n^2 \log^2 n)$. This follows exactly as in the proof of Theorem 4.2. Indeed, we can argue that if $V(t') = V(t)$ at some time $t' > t$, this means that a pair of nodes performing a random walk according to the original process have not met between times t and t'. Upper bounding the latter using the union bound, we once again obtain that the expected time it takes for $V(t)$ to shrink by 1 is the expected meeting time in the virtual process times a multiplicative factor of $O(\log n)$. □

4.4 Expected Convergence Time over Specific Networks

In this section, we provide some improved bounds for the expected convergence time of the quantized Metropolis protocol on some concrete networks. In several cases we obtain bounds which are an order-of-magnitude better than the essentially quadratic convergence times derived in the previous sections.

Our main technical tool will be to rely on the analogy between hitting times and expected resistances in the network. Indeed, recall that the quadratic bounds obtained in the earlier section were all consequences of the result from [5] that hitting times in the Metropolis chain are $O(n^2)$ on any graph. To obtain improved bounds, we only need to get improved upper bounds on hitting times. For example, if we show that the hitting times in a certain class of graphs are $O(n)$, then repeating our proof of Theorem 4.2 verbatim we would obtain an $O(n \log n)$ bound on the convergence time of the Metropolis quantized consensus protocol on these graphs. To obtain these improved bounds on hitting times, we will make use of the analogy between hitting times and electrical resistances, described next.

Suppose $Q \in \mathbb{R}^{n\times n}$ is the transition matrix of a reversible random walk with stationary probability $\pi(\cdot)$. The conductance $c(x, y)$ is defined as $c(x, y) = \pi(x)Q(x, y)$ and $c(x) := \sum_{y\in N(x)} c(x, y)$. Note that if $Q(x, y) = 0$, then $c(x, y) = 0$; furthermore, by reversibility of Q, we have $c(x, y) = c(y, x)$. Therefore, one may think of $c(\cdot)$ as a function on the "edges" $e = (x, y)$ of the graph corresponding to Q. The resistance of each edge e is defined to be the inverse of conductance, i.e. $r(e) = \frac{1}{c(e)}$. We define $\mathcal{R}(a \leftrightarrow b)$ to be the resistance between nodes a and b if each edge in the graph has resistance $r(e)$. Finally, $H^{\mathrm{Q}}(x, y)$ is defined to be the hitting time from x to y in the Markov chain corresponding to Q.

Lemma 4.6 [Proposition 10.6, [9]] *Under the above assumptions,*

$$H^{\mathrm{Q}}(a, b) + H^{\mathrm{Q}}(b, a) = c_{\mathcal{G}}\mathcal{R}(a \leftrightarrow b),$$

where $c_{\mathcal{G}} = \sum_{x\in\mathcal{V}} c(x)$.

For the special case of the Metropolis chain, we can specialize Lemma 4.6 to obtain the following corollary.

Corollary 4.1 *If Q is the Metropolis matrix on the graph $\mathcal{G}$, i.e., $Q = \mathcal{M}$, then*

$$H^{\mathrm{m}}(a, b) + H^{\mathrm{m}}(b, a) = \mathcal{R}(a \leftrightarrow b),$$

Proof Since $\mathcal{M}$ is symmetric, it is reversible and we may apply Lemma 4.6. Moreover, $\pi(i) = 1/n$. Defining $N(i)$ to be the set of nodes j with $\mathcal{M}_{ij} \neq 0$, the corollary now follows by arguing that

$$c_{\mathcal{G}} = \sum_{i=1}^{n} \sum_{j\in N(i)} \frac{1}{n}\mathcal{M}_{ij} = \frac{n}{n} = 1.$$

□

Corollary 4.1 provides us a technique to obtain improved convergence times for Metropolis consensus on some specific graphs. Indeed, by repeating the proof of Theorem 4.2 verbatim, we obtain that the convergence time of quantized Metropolis consensus is

$$\left(\max_{a,b\in\mathcal{V}} R(a \leftrightarrow b)\right) O(\log n)$$

where $R(a \to b)$ is the resistance between a and b in the graph where the edge (i, j) has resistance $n \max(d(i), d(j))$. We next apply this fact to a variety of graphs.

- *Line Graph*
  For the case of line graph, we always have $\max\{d(x), d(y)\} = 2, \forall\{x, y\} \in \mathcal{E}$. Therefore, in Corollary 4.1, the resistance of every edge of the electrical circuit associated with the Metropolis chain for the line graph is $2n$. Hence $\max_{a,b\in\mathcal{V}} \mathcal{R}(a \leftrightarrow b) = 2n(n-1)$. In particular, $\max_{a,b\in\mathcal{V}} H^{\mathrm{m}}(a, b) = O(n(n-1))$. The bound we obtain in this case is essentially identical to the bound of Theorem 4.2.
- *2-Dimensional Grids*
  A well-known result for the 2-dimensional grids (Proposition 9.16, [9]) states that the largest resistance between any pair of nodes of an $n \times n$ grid where each edge has a unit size resistance is bounded from above by $2\log(n)$. For the case of Metropolis chain over $2D$ grid, $\max\{d(x), d(y)\} \leq 4, \forall\{x, y\} \in \mathcal{G}$. By Corollary 4.1, this allows us to bound hitting times in the 2D grid as $O(n \log n)$. We thus obtain a convergence time bound of $O(n \log^2(n))$ for the quantized Metropolis protocol, which is better than the bound of Theorem 4.2 by a multiplicative factor of $n/\log n$.
- *3-Dimensional Grids*
  For 3-dimensional grids it is known that the largest resistance of an $n \times n \times n$ grid with unit resistances on each edge is bounded from above by a universal constant independent of n. Using the same line of argument as $2D$ grids, one can see that $\max\{d(x), d(y)\} \leq 6$, which in turn means that the maximum expected hitting time of the Metropolis chain over $3D$ grids is at most $O(n)$. We thus obtain a convergence time bound of $O(n \log(n))$ for the quantized Metropolis protocol, which is better than the bound of Theorem 4.2 by a multiplicative factor of n.
- *Star Graph*
  For the star graph (a tree with a root and $n-1$ leaves), we have $\max\{d(x), d(y)\} = n-1, \forall\{x, y\} \in \mathcal{G}$. This shows that $\max_{a,b\in\mathcal{V}} \mathcal{R}(a \leftrightarrow b) = n(n-1)$. This leads to bounds which are essentially the same as the bounds of Theorem 4.2.
  This may appear somewhat counter-intuitive at first glance, since the simple random walk on the star graph is easily seen to have a linear hitting time. However, the Metropolis chain attains symmetry (which is crucial for consensus) by putting large self-loops at the leaves, which leads to a much slower hitting time.
- *Lollipop Graph*
  For the lollipop graph (two clusters of size $\lfloor\frac{n}{3}\rfloor$ which are connected by a line graph of $n - 2\lfloor\frac{n}{3}\rfloor$ nodes), one can argue that $\max\{d(x), d(y)\} \leq \lfloor\frac{n}{3}\rfloor$ for the edges belonging to the clusters and $\max\{d(x), d(y)\} \leq 2$ for the edges belonging to the path. This implies that $\max_{a,b\in\mathcal{V}} R(a \leftrightarrow b) = O(n^2)$, once again leading to bounds which are identical to those of Theorem 4.2.
- *Random Geometric Graph*
  A random geometric graph $\mathcal{G}(n, r)$ is a graph which is obtained by distributing n nodes uniformly on the unit square and connecting two nodes if and only if the Euclidean distance between these two nodes is at most r. Such graphs are popular in modeling wireless ad-hoc and sensor networks [10].

It was shown in [10] (Lemma 2.13 and Sects. 2.3–2.5.2) that for $c > 1$, $r^2 = c8\log n$ is sufficient to guarantee that with high probability $\mathcal{G}(n, r)$ is "geo-dense," roughly meaning that the geometric graph has almost uniform node density across the unit square. In particular, for $r^2 \geq c8\log n$ all nodes have degree of $O(nr^2)$ with high probability. Furthermore, for this range of r and by Theorem 2.20 in [10], the maximum resistance between two nodes of a geometric graph when each node has a unit resistance is $O(\frac{1}{nr^2})$ with high probability. Putting these two results together and in view of Corollary 4.1, we can see that for a random geometric graph with $r^2 \geq c8\log n,\ c > 1$ we have $\max_{a,b} H(a, b) = O(n \times nr^2 \times \frac{1}{nr^2}) = O(n)$. Therefore, the expected convergence time of the quantized Metropolis protocol for this range of r is at most $O(n\log n)$ with high probability.

- *Erdos-Renyi Graphs*
  For Erdos-Renyi graphs $\mathcal{G}(n, p)$, where each edge appears with probability p independent of the other edges, the degree of each vertex is highly concentrated around its mean, i.e., $(n-1)p$. In fact, it can be shown that for $p = \frac{\log n}{n}\omega(n)$ where $\omega(n) \to \infty$ arbitrarily slowly, the ratio of degree of each vertex x to its mean approaches 1, i.e., $\frac{d(x)}{(n-1)p} \to 1$, almost always (Theorem 8.5.1, [11]). This means that for sufficiently large n, with high probability $\max\{d(x), d(y)\} = O((n-1)p)$. Moreover, it is known that for $p \in (\frac{\log n}{n}, 1]$, the diameter of $G(n, p)$ is at most $O(\log n)$ almost always. Putting these two results together and in view of Corollary 4.1, we can see that for the Erdos-Renyi graph with $p = \frac{\log n}{n}\omega(n)$ we have $\max_{a,b} H(a, b) = O(n \times (n-1)p \times \log n) = O((n-1)\log^2 n\ \omega(n))$ (note that $\omega(n)$ can grow arbitrarily slowly to infinity, e.g., $\omega(n) := \log(\log(n))$). Therefore, the expected convergence time of the quantized Metropolis protocol for this range of $p = \frac{\log n}{n}\omega(n)$ is at most $O((n-1)\log^2 n\ \omega(n))O(\log n) = O((n-1)\log^3 n\ \omega(n))$ with high probability.

4.5 Simulations

In this section, we report on the results of some simulations on the performance of the quantized Metropolis method. Since the key bottleneck in performance is the expected time until a nontrivial update takes place, here we compare this time to the quadratic upper bounds we have derived for a variety of graphs.

More precisely, we consider the ratio of $\frac{T}{n^2}$ for four simple classes of graphs $\mathcal{G}$, namely, line, star, lollipop and 3-regular graphs, as well as Erdos-Renyi graphs. We have depicted our simulation results in Fig. 4.4 while we let the number of nodes vary from 1 to 70 for each of these graphs.

For the case of the Erdos-Renyi graph, we assume that the probability that each edge appears in the graph is $\frac{1}{2}$ and independent of the other edges. As seen in Fig. 4.4, the ratio of $\frac{T}{n^2}$ is bounded from above by 0.5 and vanishes very fast, corresponding to a sub-quadratic convergence time. For the case of the lollipop graph, we consider two cliques, each of size $\lceil \frac{n}{4} \rceil$, which are connected by a path of the remaining nodes.

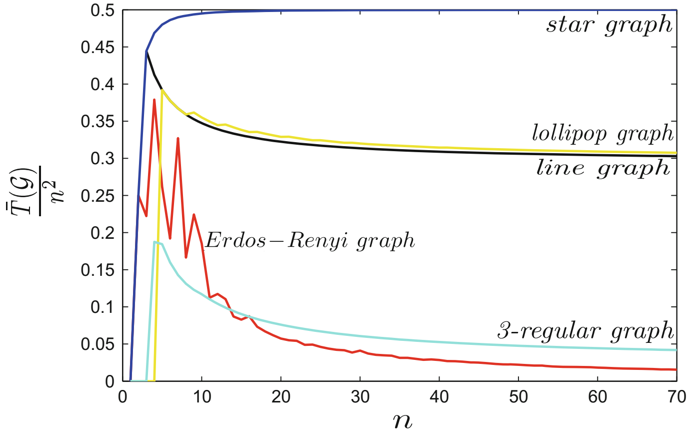


Fig. 4.4 Comparison between the maximum expected time that a nontrivial update takes place and the upper bound of n^2 for static networks

As seen again in Fig. 4.4, the ratio of $\frac{\overline{T}}{n^2}$ for each of the line graph, lollipop and star graph is bounded from above by 0.5 and asymptotically converges to a constant, meaning that our quadratic upper bounds are essentially sharp for these graphs.

Finally, in the case of 3-regular graphs, we consider a cycle of n nodes where each node is connected to its diagonally opposite node on the cycle (this is possible when n is an even number, and when n is odd we leave the final node n to be of degree 2). Again, it can be seen from Fig. 4.4 that the quantized Metropolis dynamics display good performance on such graphs.

Next we consider the expected time for the occurrence of a nontrivial update over various sequences of time-varying graphs. For this purpose, we consider five different sequences of time-varying networks, all having the property that graph changes can only occur at discrete time instances $k = 1, 2, \ldots$. In particular, we consider:

- (I) An alternating sequence of line and star networks, i.e.,

$$\mathcal{G}(k) = \begin{cases} \text{line graph,} & \text{if } k \text{ is even} \\ \text{star graph,} & \text{if } k \text{ is odd.} \end{cases}$$

- (II) An alternating sequence of line, star and lollipop networks, i.e.,

$$\mathcal{G}(k) = \begin{cases} \text{lollipop graph,} & \text{if } \mathrm{mod}(k, 3) = 0 \\ \text{line graph,} & \text{if } \mathrm{mod}(k, 3) = 1 \\ \text{star graph,} & \text{if } \mathrm{mod}(k, 3) = 2. \end{cases}$$

- (III) A random sequence of line and star networks generated by tossing an unbiased coin, i.e.,

$$\mathcal{G}(k) = \begin{cases} \text{line graph}, & \text{w.p. } \frac{1}{2} \\ \text{star graph}, & \text{w.p. } \frac{1}{2}. \end{cases}$$

- (IV) A sequence of Erdos-Renyi graphs with $\frac{1}{2}$ probability of emerging an edge in each graph.
- (V) A uniformly at random generated sequence of line, star, lollipop and Erdos-Renyi graphs, i.e.,

$$\mathcal{G}(k) = \begin{cases} \text{line graph}, & \text{w.p. } \frac{1}{4} \\ \text{lollipop graph}, & \text{w.p. } \frac{1}{4} \\ \text{star graph}, & \text{w.p. } \frac{1}{4} \\ \text{Erdos-Renyi graph}(p = \frac{1}{2}), & \text{w.p. } \frac{1}{4}. \end{cases}$$

In each of the above cases we let the number of nodes vary from 1 to 50. We let the length of the sequence of the time-varying graphs be large enough such that in all the sample paths we generate, at least one nontrivial update takes place (this ended up resulting in a length of 5,000). We ran the quantized Metropolis protocol over these sequences of time-varying networks 20,000 times and took an average of the waiting times for the occurrence of a nontrivial update. We initialized the process with an initial condition of two nodes having values 0 and 2, while all the rest have value 1; furthermore, we simulated this for every possible pair of starting nodes for the values 0 and 2 and take the maximum. The results are depicted in the Fig. 4.5.

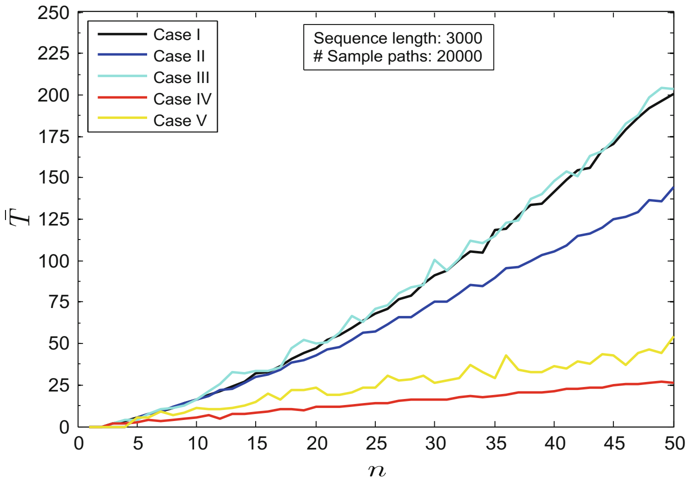


Fig. 4.5 Comparison between the maximum expected time that a nontrivial update takes place and the upper bound of n^2 for time-varying networks

In the first three cases (I, II and III), the curve shows a quadratic behavior in the number of nodes, matching the bounds of Theorem 4.3. For the two other cases (IV, V), the curve appears to be linear, once again confirming the fast performance of Metropolis consensus over Erdos-Renyi graphs.

4.6 Conclusion

We have studied the quantized consensus problem on undirected connected networks in both static and time-varying settings. Using Metropolis chains within the structure of quantized consensus protocols, we were able to improve the expected convergence time relative to the previous works in the literature. In particular, we have proved an upper bound of $O(n^2 \log n)$ for the convergence time of quantized Metropolis dynamics over static networks and an upper bound of $O(n^2 \log^2 n)$ over dynamic networks.

References

1. T. Başar, S. R. Etesami, and A. Olshevsky, "Convergence time of quantized Metropolis consensus over time-varying networks," *IEEE Transactions on Automatic Control, vol 61, December*, pp. 4048–4054, 2016.
2. A. Kashyap, T. Başar, and R. Srikant, "Quantized consensus," *Automatica*, vol. 43, no. 7, pp. 1192–1203, 2007.
3. D. Aldous and J. Fill, *Reversible Markov chains and random walks on graphs*. Book in preparation - preprint available at http://www.stat.Berkeley.EDU/users/aldous, 2002.
4. D. Coppersmith, P. Tetali, and P. Winkler, "Collisions among random walks on a graph," *SIAM Journal on Discrete Mathematics*, vol. 6, pp. 363–374, 1993.
5. Y. Nonaka, H. Ono, K. Sadakane, and M. Yamashita, "The hitting and cover times of Metropolis walks," *Theoretical Computer Science*, vol. 411, no. 16, pp. 1889–1894, 2010.
6. D. Fremlin, *Measure Theory*. ISBN 978-0-9538129-7-4, 2010, vol. 2.
7. E. Seneta, *Non-negative Matrices and Markov Chains*. Springer, 2006.
8. D. G. Feingold, R. S. Varga et al., "Block diagonally dominant matrices and generalizations of the Gerschgorin circle theorem," *Pacific J. Math*, vol. 12, no. 4, pp. 1241–1250, 1962.
9. D. A. Levin, Y. Peres, and E. L. Wilmer, *Markov Chains and Mixing Times*. American Mathematical Society, 2009.
10. C. Avin, "Random geometric graphs: an algorithmic perspective," Ph.D. dissertation, University of California Los Angeles, 2006.
11. N. Alon and J. H. Spencer, *The Probabilistic Method*. John Wiley & Sons, 2004.

Chapter 5
Hegselmann-Krause Opinion Dynamics in Finite Dimensions

In this chapter we describe the discrete-time Hegselmann-Krause opinion dynamics model as introduced in [1], and study its termination time under various scenarios. Despite many works and efforts toward understanding the nature of such dynamics, still a rigorous analysis of the Hegselmann-Krasue dynamics is both theoretically and practically important. The theoretical aspect is that it allows us to develop novel tools useful to study more complex time and state dependent evolutionary dynamics and elaborate on their connections with other fields. The practical aspect is that, other than applications in the modeling of opinion dynamics in social networks, the model has applications in the robotics rendezvous and engineering, linguistic formation, modeling of human perception and cognition, behavioral studies and modeling in economics, psychology, and political science [2–4].

In the Hegselmann-Krause model, a finite number of agents frequently update their opinions based on the possible interactions among them. While Hegselmann-Krause dynamics can be considered as a special class of averaging dynamics, what makes the analysis of such dynamics more complicated but interesting is the strong coupling between the network structure and the opinion evolution (state) of these dynamics. In other words, Hegselmann-Krause dynamics belong to the class of time-dependent, state dependent, switching dynamics. Moreover, as opposed to many earlier models for averaging dynamics where the goal is to drive the entire system to some consensus point, Hegselmann-Krasue dynamics do not necessarily converge to the same opinion for all the agents, which introduces a new concept of opinion polarization in social networks. In fact, in many situations such as political elections and product marketings when there are multiple candidates or product choices to be selected among, there is neither a desire for consensus nor any tendency for the underlying process to approach a common outcome, which makes the analysis of the Hegselmann-Krause opinion dynamics more appealing as an alternative to study disagreement beside consensus.

S.R. Etesami, *Potential-Based Analysis of Social, Communication, and Distributed Networks*, Springer Theses, DOI 10.1007/978-3-319-54289-8_5

5.1 Hegselmann-Krause Dynamics

Let us assume that we have a set of n agents $[n] = \{1, \ldots, n\}$ and we want to model the interactions among their opinions. It is assumed that at each time $t = 0, 1, 2, \ldots$, the opinion of agent $i \in [n]$ can be represented by a vector $x_i(t) \in \mathbb{R}^d$ for some $d \geq 1$. According to that model, the evolution of opinion vectors can be modeled by the following discrete-time dynamics:

$$x(t+1) = A(t, x(t), \vec{\epsilon})x(t), \tag{5.1}$$

where $A(t, x(t), \vec{\epsilon})$ is an $n \times n$ row-stochastic matrix and $x(t)$ is the $n \times d$ matrix such that its ith row contains the opinion of the ith agent at time $t = 0, 1, 2, \ldots$, i.e., it is equal to $x_i(t)$. We refer to $x(t)$ as the *opinion profile* at time t. The entries of $A(t, x(t), \vec{\epsilon})$ are functions of time step t, current profile $x(t)$, confidence vector $\vec{\epsilon} = (\epsilon_1, \epsilon_2, \ldots, \epsilon_n) > 0$ and an updating scheme. The parameters $\epsilon_i, i \in [n]$ are referred to as the ***confidence bounds***. In the homogeneous case of the dynamics, we assume that $\epsilon_i = \epsilon,\ \forall i \in [n]$ for some $\epsilon > 0$, while in the heterogeneous model, different agents may have different bounds of confidence. Our focus in this chapter is mainly on the homogeneous model, but we also analyze the heterogeneous case toward the end, in Sect. 5.4. For the sake of simplicity of notation and for a fixed $x(0) \in \mathbb{R}^{n\times d}$, we drop the dependency of $A(t, x(t), \vec{\epsilon})$ on $x(t)$ and ϵ and simply write $A(t)$. In what follows next, we distinguish two different versions of Hegselmann-Krause dynamics.

5.1.1 Synchronous Hegselmann-Krause Model

In the synchronous Hegselmann-Krause model, each agent i updates its value at time $t = 0, 1, 2, \ldots$, by averaging its own value and the values of all the other agents that are in its ϵ-neighborhood at time t. To be more specific, given a profile $x(t)$ at time t, define the matrix $A(t)$ in (5.1) by:

$$A_{ij}(t) = \begin{cases} \frac{1}{|\mathcal{N}_i(t)|} & \text{if } j \in \mathcal{N}_i(t), \\ 0 & \text{else,} \end{cases} \tag{5.2}$$

where $\mathcal{N}_i(t)$ is the set of agents in the ϵ-neighborhood of agent i, i.e.,

$$\mathcal{N}_i(t) = \{j \in [n] \mid \|x_i(t) - x_j(t)\| \leq \epsilon\}.$$

5.1.2 Asynchronous Hegselmann-Krause Model

In the asynchronous case and at each time instant $t = 0, 1, 2, \ldots$, only one agent, namely i^*, updates its value to the average of its neighbors, while the others remain unchanged. Selection of such an agent may be at random or based on some predefined order. In this section, we assume that the agents are chosen uniformly at random to update their opinions. In that case the updating matrix $A(t, x(t), \vec{\epsilon})$ given in (5.1) can be written as

$$A_{ij}(t) = \begin{cases} \frac{1}{|\mathcal{N}_{i^*}(t)|} & \text{if } i = i^*, j \in \mathcal{N}_{i^*}(t), \\ 1 & \text{if } i = j \neq i^* \\ 0 & \text{else,} \end{cases} \tag{5.3}$$

where here we have assumed that agent i^* updates its opinion at time t.

Remark 6 In the heterogeneous Hegselmann-Krause model, each agent i is able to observe only its ϵ_i-neighborhood, and we have

$$\mathcal{N}_i(t, \epsilon_i) = \{j \in [n] \mid \|x_i(t) - x_j(t)\| \leq \epsilon_i\}.$$

Remark 7 There are other types of Hegselmann-Krause dynamics where the evolution of dynamics is subject to noise or perturbation in the system or when the agents are truth seekers in the sense that they are attracted by the truth by a positive amount [5, 6]. Moreover, the continuous version of the Hegselmann-Krause model, in which a continuum of opinions are involved in the dynamics, has been considered in [7–9].

Remark 8 As can be seen from the above formulations, the Hegselmann-Krause dynamics do not preserve the opinion average of the agents, and the evolution of the system strongly depends on the history and the states, which may switch between different topologies. In fact, it is not possible to determine the topology of the network at the current time, unless one can observe the state of the system in the previous time step. Those facts make the analysis of such dynamics much more complicated than analysis of the average-preserving dynamics with fixed topology.

5.2 Synchronous Multidimensional Hegselmann-Krause Dynamics

In this section we consider the homogeneous synchronous Hegselmann-Krause model as was introduced in (5.2). In fact, Fig. 5.1 shows an example of evolution of such dynamics in two dimensions when the agents are initially positioned around a unit circle. Similarly, Fig. 5.2 illustrates another example of evolution of such dynamics in three dimensions when some of the agents are initially positioned around a

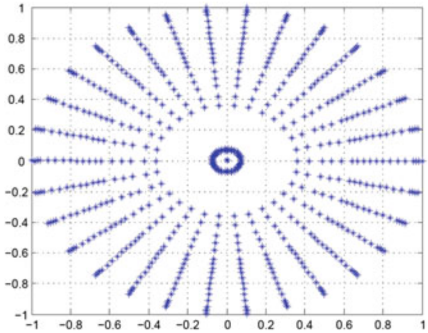


Fig. 5.1 $d = 2$, $n = 30$. In the *left* figure $\epsilon = 0.4$, and we run the dynamics $t = 50$ times, while in the *right* figure $\epsilon = 0.55$ and we run the dynamics for only $t = 10$ iterations

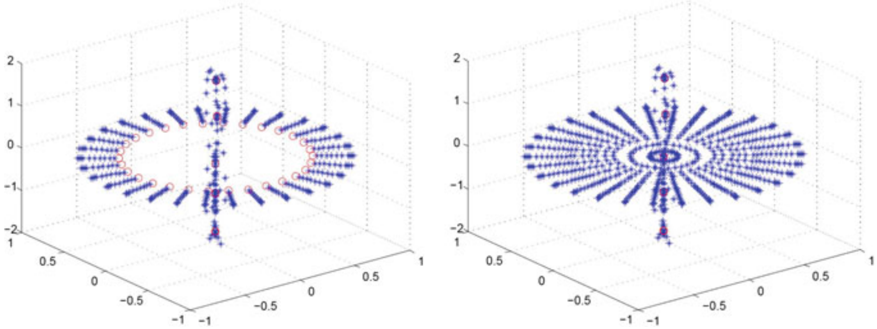


Fig. 5.2 $d = 3$, $n = 60$ and $\epsilon = 0.4$. In the *left* and *right* figures the dynamics are run for $t = 10$ and $t = 50$ iterations, respectively

unit circle parallel to the $X - Y$ plane, and the rest of agents are positioned along the Z-axis using a *sinc* distribution. In Fig. 5.2, the red points in the left and right side represent the location of the agents after 10 and 50 iterations, respectively.

In fact, one of the fundamental concepts and properties of the synchronous Hegselmann-Krause dynamics that will be used extensively throughout this chapter is that the dynamics admit a quadratic Lyapunov function [8, 10–12].

We start our discussion by introducing some notation that will be used throughout this section.

Definition 5.1 We say that a time instance t is a *merging time* for the dynamics if two agents with different opinions move to the same place.

Based on that definition, we can see that if two agents i and j merge at time instant t, then they will have the same opinion at time $t + 1$ and onward, while their common opinion may vary with time. Moreover, prior to the termination time of the dynamics, we cannot have more than n merging times, since there are n agents in the model. In what follows next, we define the notions of termination time and communication graphs.

Definition 5.2 For every set of $n \geq 1$ agents we define the *termination time* T_n of the synchronous Hegselmann-Krause dynamics to be the maximum number of iterations before steady state is reached over all the initial profiles.

Definition 5.3 Given an opinion profile at time t, we associate with that opinion profile an undirected graph $\mathcal{G}(t) = ([n], \mathcal{E}(t))$ where the edge $(i, j) \in \mathcal{E}(t)$ if and only if $i \in \mathcal{N}_j(t)$. We refer to such a graph as the *communication graph* or *communication topology* of the dynamics at time step t. Furthermore, a connected component of the communication graph is called δ-*trivial* for some $\delta > 0$, if all the agents in that component lie within a distance of at most δ from each other.

Remark 9 From Definition 5.3, it is not hard to see that for any $\delta < \epsilon$, a δ-trivial component forms a complete component (clique) in the communication topology of the dynamics. In particular, if there is such a δ-trivial component at some time t, then in the next time step, all the agents in that component will merge to the same opinion.

In our earlier work [13], we were able to analyze the termination time of the Hegselmann-Krause dynamics based on the number of isolated agents throughout the dynamics.

Theorem 5.1 *For the termination time T_n of the synchronous Hegselmann-Krause dynamics in $\mathbb{R}^d$, we have:*

$$\sum_{t=0}^{T_n} (\frac{1}{2})^{|S_0(t)|} < 8n^6,$$

where $S_0(t)$ denotes the set of agents (singletons) at time t who do not observe any opinions other than those inside their neighborhood, i.e., $i \in S_0(t)$ if and only if $\mathcal{N}_i(t) = \{x_i(t)\}$.

Proof A proof can be found in [13]. □

As a particular result of Theorem 5.1, if for a particular instance of the dynamics, the agents maintain the connectivity throughout the dynamics, we conclude that $T_n = O(n^6)$. In fact, Theorem 5.1 gives us the idea that the termination time of the dynamics greatly depends on the connectivity of the underlying graph and should be independent of the dimension of the opinions (d). In this thesis, we resolve that problem and show that indeed, the termination time is independent of the dimension. That answers one of the open questions raised in [14] related to the existence of a tighter polynomial upper bound independent of the dimension of the opinion space. For that purpose, we utilize a quadratic Lyapunov function that was introduced earlier in [12].

Lemma 5.1 *Let* $V(t) = \sum_{i,j\in[n]} \min\{\|x_i(t) - x_j(t)\|^2, \epsilon^2\}$. *Then* $V(\cdot)$ *is non-increasing along the trajectory of the synchronous Hegselmann-Krause dynamics. In particular, we have*

$$V(t) - V(t+1) \geq 4\sum_{\ell=1}^{n} \|x_\ell(t+1) - x_\ell(t)\|^2.$$

Proof A proof can be found in [12]. □

In the following theorem, we provide a lower bound for the amount of decrease of the above Lyapunov function as long as there exists one non-ϵ-trivial component in the dynamics. This is the first analytic result which connects the convergence speed of such dynamics to the spectrum of the adjacency matrix of the connectivity graph in the Hegselmann-Krause dynamics.

Theorem 5.2 *The termination time of the synchronous Hegselmann-Krause dynamics in arbitrary finite dimensions is independent of the dimension and is bounded from above by* $T_n \leq n^8 + n$.

Proof Let us assume that the opinion profile $x(t) = (x_1(t), x_2(t), \dots, x_n(t))'$ is not an equilibrium point of the dynamics and that time t is not a merging time. Therefore, without loss of generality, we may assume that the communication graph at time t is connected with a non-ϵ-trivial component; otherwise, we can restrict ourselves to one of the non-ϵ-trivial components. (Note that such a non-ϵ-trivial component exists, because of Remark 9 and the fact that t is not a merging time.) By projecting each individual column of $x(t)$ to the consensus vector $\mathbf{1}$ we can write

$$x(t) = \Big[c_1\mathbf{1}|c_2\mathbf{1}|\dots|c_d\mathbf{1}\Big] + \Big[\bar{c}_1\overset{\perp}{\mathbf{1}}^{(1)}|\bar{c}_2\overset{\perp}{\mathbf{1}}^{(2)}|\dots|\bar{c}_d\overset{\perp}{\mathbf{1}}^{(d)}\Big], \tag{5.4}$$

where $\overset{\perp}{\mathbf{1}}^{(k)}, k = 1, \dots, d$ are column vectors of unit size that are orthogonal to the consensus vector, i.e., $\mathbf{1}'\overset{\perp}{\mathbf{1}}^{(k)} = 0$, and $c_k, \bar{c}_k, k = 1, \dots, d$ are coefficients of projection of the kth column of $x(t)$ on $\mathbf{1}$ and $\overset{\perp}{\mathbf{1}}^{(k)}$, respectively.

Now we claim that $\sum_{k=1}^{d} \bar{c}_k^2 > \frac{\epsilon^2}{4}$. Otherwise, we show that every two agents $x_i(t)$ and $x_j(t)$ must lie within a distance of at most ϵ from each other, which is in contrast with the assumption that the component is a non-ϵ-trivial component. In fact, if $\sum_{k=1}^{d} \bar{c}_k^2 \leq \frac{\epsilon^2}{4}$, we can write

$$\begin{aligned}\|x_i(t) - x_j(t)\|^2 &= \sum_{k=1}^{d} \bar{c}_k^2\big(\overset{\perp}{\mathbf{1}}_i^{(k)} - \overset{\perp}{\mathbf{1}}_j^{(k)}\big)^2 \leq 2\sum_{k=1}^{d} \bar{c}_k^2\big((\overset{\perp}{\mathbf{1}}_i^{(k)})^2 + (\overset{\perp}{\mathbf{1}}_j^{(k)})^2\big)\\ &\leq 2\sum_{k=1}^{d} \bar{c}_k^2\big(\|\overset{\perp}{\mathbf{1}}^{(k)}\|^2 + \|\overset{\perp}{\mathbf{1}}^{(k)}\|^2\big) = 4\sum_{k=1}^{d} \bar{c}_k^2 \leq \epsilon^2,\end{aligned} \tag{5.5}$$

where the first equality is due to the decomposition given in (5.4) and the second equality is valid since the vectors $\overset{\perp}{\mathbf{1}}^{(k)}$, $k = 1\ldots, d$, are of unit size. The contradiction shows that $\sum_{k=1}^{d} \bar{c}_k^2 > \frac{\epsilon^2}{4}$.

Next, we notice that $x(t+1) = A(t)x(t)$, where $A(t)$ is the stochastic matrix defined in (5.2). Using (5.4) we can write,

$$x(t) - x(t+1) = (I - A(t))x(t) = \left[\bar{c}_1(I - A(t))\overset{\perp}{\mathbf{1}}^{(1)}|\ldots|\bar{c}_d(I - A(t))\overset{\perp}{\mathbf{1}}^{(d)}\right], \tag{5.6}$$

where the equality holds since $\mathbf{1}$ belongs to the null space of $I - A(t)$. In particular, we have

$$\begin{aligned}\sum_{\ell=1}^{n} \|x_\ell(t) - x_\ell(t+1)\|^2 &= \sum_{\ell=1}^{n}\sum_{k=1}^{d}\big(x_{\ell k}(t) - x_{\ell k}(t+1)\big)^2\\ &= \sum_{k=1}^{d}\Big(\sum_{\ell=1}^{n}\big(x_{\ell k}(t) - x_{\ell k}(t+1)\big)^2\Big)\\ &= \sum_{k=1}^{d}\bar{c}_k^2\|(I - A(t))\overset{\perp}{\mathbf{1}}^{(k)}\|^2, \end{aligned}\tag{5.7}$$

where in the last equality we have used (5.6). Let us assume that $Q(t) = (I - A(t))'(I - A(t))$. It is not hard to see that $Q(t)$ is a positive semidefinite matrix. Moreover, 0 is an eigenvalue of Q with multiplicity one, corresponding to the eigenvector $\mathbf{1}$. To see that, let us assume that there exists another vector v, such that $Q(t)v = 0$. Multiplying that equality from the left by v', we get $\|(I - A(t))v\|^2 = 0$, and hence $(I - A(t))v = 0$. Since by the Perron-Frobenius lemma (Lemma 2.5), $\mathbf{1}$ is the only unit eigenvector of $I - A(t)$ corresponding to eigenvalue 0, we conclude that $v = \alpha\mathbf{1}$ for some $\alpha \in \mathbb{R}$. In other words, $\mathbf{1}$ is the only unit eigenvector of $Q(t)$ corresponding to eigenvalue 0. Moreover, $Q(t)$ is a symmetric real-valued matrix and, hence, diagonalizable, where $\mathbf{1}$ is its only eigenvector corresponding to eigenvalue 0. That shows that the multiplicity of the eigenvalue 0 in $Q(t)$ is exactly one.

Let us use $\lambda_2(Q(t))$ to denote the second smallest eigenvalue of $Q(t)$. By the above argument, it must be strictly positive. Using the Courant-Fischer lemma (Lemma 2.7), we get $\lambda_2(Q(t)) = \min_{\|y\|=1, y\perp\mathbf{1}} y'Q(t)y$. Now for every $k = 1, \ldots, d$, we can write

$$\begin{aligned}\|(I - A(t))\overset{\perp}{\mathbf{1}}^{(k)}\|^2 &= (\overset{\perp}{\mathbf{1}}^{(k)})'(I - A(t))'(I - A(t))\overset{\perp}{\mathbf{1}}^{(k)}\\ &= (\overset{\perp}{\mathbf{1}}^{(k)})'Q(t)\overset{\perp}{\mathbf{1}}^{(k)} \geq \min_{\substack{\|y\|=1\\ y\perp\mathbf{1}}} y'Q(t)y = \lambda_2(Q(t)),\end{aligned}\tag{5.8}$$

where the inequality holds, since $\mathbf{1}'\overset{\perp}{\mathbf{1}}^{(k)} = 0$ and $\|\overset{\perp}{\mathbf{1}}^{(k)}\| = 1$. Substituting (5.8) in (5.7) we get

$$\sum_{\ell=1}^{n} \|x_\ell(t) - x_\ell(t+1)\|^2 \geq \sum_{k=1}^{d} \lambda_2(Q(t))\bar{c}_k^2 \geq \lambda_2(Q(t))\frac{\epsilon^2}{4}. \tag{5.9}$$

Henceforth, we bound $\lambda_2(Q(t))$ from below based on a function of n. For that purpose, let us assume that $D(t) = diag\big(1 + d_1(t), 1 + d_2(t), \ldots, 1 + d_n(t)\big)$, i.e., $D(t)$ is a diagonal matrix with $D_{kk}(t) = 1 + d_k(t), k \in [n]$. Moreover, let $\mathcal{L}(t)$ denote the Laplacian matrix of the communication graph at time step t. By entrywise comparison of both sides, it is not hard to see that $I - A(t) = D(t)^{-1}\mathcal{L}(t)$. Now we can write

$$\lambda_2(Q(t)) = \lambda_2\left((D(t)^{-1}\mathcal{L}(t))'(D(t)^{-1}\mathcal{L}(t))\right) = \lambda_2\left(\mathcal{L}(t)D(t)^{-2}\mathcal{L}(t)\right), \tag{5.10}$$

where the last equality is due to the fact that $\mathcal{L}(t)$ and $D(t)$ are both symmetric matrices. Next, using the same argument as above, we notice that since $\mathcal{L}(t)D(t)^{-2}\mathcal{L}(t)$ is a symmetric and real-valued matrix, it is diagonalizable, and its zero eigenvalue corresponding to eigenvector $\mathbf{1}$ has multiplicity one. To see that, let us assume that there is another vector u such that $\mathcal{L}(t)D(t)^{-2}\mathcal{L}(t)u = 0$; then, we must have

$$0 = u'\mathcal{L}(t)D(t)^{-2}\mathcal{L}(t)u = \sum_{i=1}^{n}\left(\frac{1}{1+d_i(t)}\right)^2 (\mathcal{L}(t)u)_i^2,$$

which results in $\mathcal{L}(t)u = 0$, or, equivalently, u is a scalar multiple of the consensus vector $\mathbf{1}$.

Now, using the Courant-Fischer lemma, we can write

$$\begin{aligned}\lambda_2\big(\mathcal{L}(t)D(t)^{-2}\mathcal{L}(t)\big) &= \min_{\substack{\|y\|=1\\ y\perp\mathbf{1}}} y'\mathcal{L}(t)D(t)^{-2}\mathcal{L}(t)y \geq \min_{\substack{\|y\|=1\\ y\perp\mathbf{1}}} y'\mathcal{L}(t)(\frac{1}{n^2}I)\mathcal{L}(t)y\\ &= \lambda_2\Big(\mathcal{L}(t)(\frac{1}{n^2}I)\mathcal{L}(t)\Big) = \frac{1}{n^2}\lambda_2\big(\mathcal{L}^2(t)\big) = \frac{1}{n^2}\lambda_2^2\big(\mathcal{L}(t)\big),\end{aligned} \tag{5.11}$$

where the last equality is due to the fact that $\mathcal{L}$ is diagonalizable (it is a symmetric and real-valued matrix) with an eigenvalue 0 of multiplicity 1. Substituting (5.11) in (5.10) we get $\lambda_2(Q(t)) \geq \frac{1}{n^2}\lambda_2^2\big(\mathcal{L}(t)\big)$. Now, using Cheeger's inequality (Lemma 2.6) and since $\mathcal{L}(t)$ is the Laplacian of a connected graph, we can bound $\lambda_2\big(\mathcal{L}(t)\big)$ from below by $\frac{2}{n^2}$, which is due to the isoperimetric number of the communication graph for the minimum cut set. Putting it all together, we have

$$\lambda_2(Q(t)) \geq \frac{1}{n^2}\lambda_2^2\big(\mathcal{L}(t)\big) \geq \frac{4}{n^6}. \tag{5.12}$$

Finally, combining (5.12) with (5.9), we conclude that the amount of decrease in the quadratic Lyapunov function if there is a non-ϵ-trivial component is at least $\frac{\epsilon^2}{n^6}$. In other words, if t is not a merging time, we have $V(t) - V(t+1) \geq \frac{\epsilon^2}{n^6}$. Since by definition $V(\cdot)$ is always a nonnegative quantity with $V(0) \leq \epsilon^2 n^2$ and the number of merging times can be at most n, we conclude that the termination time is bounded from above by $n^8 + n$. □

Remark 10 The sum of the terms $\sum_{\ell=1}^{n} \|x_\ell(t) - x_\ell(t+1)\|^2$ which appear in (5.7) is a measure of decrease for the Lyapunov function given for the Hegselmann-Krause dynamics. This quantity, which is known as ***total-s-energy***,is an important measure which may arise in analysis of various multi-agent systems, see, e.g., [15]. In fact, the above proof provides an analytic tool to connect the total-s-energy of a multi-agent system to the spectral gap of its underlying dynamics.

Here, it is worth to make a comparison between the result of Theorem 5.2 and those in the typical averaging dynamics. In the classic averaging dynamics, either the network topology is fixed and the nodes update their opinions based on some updating rule, or the networks are time-varying but their evolution is somehow *independent* from the opinion updates. In both of these cases using standard techniques from Markov chains or graph theory one can show that the speed of convergence to the final outcome is exponentially fast with a rate proportional to the multiplication of the second smallest eigenvalues of the Laplacian matrices at different time instances, i.e., $\prod_t \lambda_2(\mathcal{L}(t))$. However, this is not the case for the Hegselmann-Krause dynamics in which the communication networks and the opinion states are highly correlated. In fact, in Hegselmann-Krause dynamics the communication network at some time defines the opinions (state) at the next time instant, and the new state determines the next communication graph and so on, such that there is no way of decoupling the opinion states and the communication networks. However, what we have shown in Theorem 5.2 is somehow interesting. It shows that despite such a strong network-state coupling, the entire system still looses energy, but the rate of the energy loss (or convergence rate to the final state) is proportional to the *sum* of the squared eigenvalues of all the components at different times $\sum_{i,t} \lambda_2^2(\mathcal{L}_{C_i}(t))$ (even though we may have disconnectivity in the communication network!).[1] As a result of such coupling we loose exponential convergence to the final outcome, but we still have a polynomial convergence time to the steady state of the dynamics. This can be viewed as a generalization of the classic averaging dynamics to highly dynamic and correlated settings.

[1] Here $\mathcal{L}_{C_i}(t)$ denotes the Laplacian matrix of the ith component C_i of the communication graph $\mathcal{G}(t)$ at time instant t.

5.3 Asynchronous Hegselmann-Krause Dynamics

In this section, we consider the asynchronous Hegselmann-Krause dynamics as introduced in Sect. 5.1. We first notice that such dynamics do not necessarily reach their steady state in finite time. The simplest case one can consider is when there are only two agents on the real line, separated by a distance less than the confidence bound ϵ. In such a case, no matter what the order of the updating process, the agents will never arrive at the same opinion or disappear from each other's neighborhood. The two agents will get closer and closer and asymptotically converge to some steady state. That justifies asymptotic analysis of the asynchronous Hegselmann-Krause dynamics, which we will consider in this section.

In fact, one can easily show that unless the dynamics start from a steady state, it will never reach its steady state in finite time for any asynchronous updating scheme. The reason is that unless the dynamics start from a steady state, at any time instant t, there are at least two agents i and j who are connected ($j \in \mathcal{N}_i(t)$), and updating any of them does not bring them to the same opinion. Furthermore, unlike the synchronous case in one dimension, where the order of agents' opinions is preserved throughout the dynamics, in the asynchronous case, the order of the agents' opinions may or may not change, depending on the updating scheme. In this section, we consider a uniformly randomized updating scheme for the agents and analyze the asymptotic convergence of such dynamics to their steady state. But before we start, we need the following two definitions.

Definition 5.4 We call an updating process a *uniform* updating scheme for the asynchronous Hegselmann-Krause mode if at each time instant $t = 0, 1, \ldots$, only one agent is chosen independently and with probability $\frac{1}{n}$ from the set of all agents $[n]$ and updates its opinion.

Definition 5.5 Given a $\delta > 0$, we say that an opinion profile $x(t)$ is a δ-equilibrium if the set of agents partition into different sets (clusters) $\{C_1, C_2, \ldots, C_m\}$ for some $m \in \mathbb{N}$ such that $dist\big(conv(C_i), conv(C_j)\big) > \epsilon, \forall i \neq j$ and $diam(conv(C_k)) < \delta, \forall k = 1 \ldots m$.

In fact, Definition 5.5 simply states that a profile $x(t)$ is a δ-equilibrium if the opinions of agents at time t form some small groups of diameter at most δ that are far from each other by a distance of at least ϵ. Next, we introduce a network formation game that can explain the behavior of the agents in asynchronous Hegselmann-Krause dynamics.

5.3.1 *Network Formation Game*

Let us consider a set of n road constructors (players) in $\mathbb{R}^d$ who are funded by the government to construct roads. The budget that the government allocates to each player at the beginning is a fixed amount and is equal to $\$(n-1)\epsilon^2$ ($\$\epsilon^2$ support for each possible road that one player can construct). Ideally, the government would like for all the possible $\binom{n}{2}$ roads to be constructed by the players. To that end and in order to create an incentive for players to build as many roads as they can, the government will punish each player by $\$\epsilon^2$ if he or she decides not to construct a road (i.e., the government will take that player's supporting $\$\epsilon^2$ back). On the other hand, each player has the ability to construct roads only within an ϵ^2-neighborhood of himself or herself. (One can assume that the players do not take risks and do not want to spend money beyond the support they received from the government per road.) In such a game, players act myopically, trying to build roads with those who are most beneficial to them. If two players who are located at $x, y \in \mathbb{R}^d$ build a road together, the cost to them is naturally proportional to their distance from each other and is equal to $\|x-y\|^2$. (The farther the players are from each other, the more costly to make a road.) Therefore, in that setting, the payoff for the ith player, $i \in [n]$, at location x_i can be formulated as

$$U_i(x_i, x_{-i}) = (n-1)\epsilon^2 - \sum_{j=1}^{n} \min\{\|x_i - x_j\|^2, \epsilon^2\}, \tag{5.13}$$

where x_{-i} denotes the actions of all players except the ith one. In such a game, we assume that agents act rationally and are able to compute and play their better response at time steps $t = 0, 1, 2, \ldots$. Based on the above scenario, we have the following lemma.

Lemma 5.2 *The sequence of the players' better responses in the network formation game under some specific updating scheme is equivalent to the evolution of the asynchronous Hegselmann-Krause dynamics under the same updating scheme.*

Proof Let us assume that at time step t the ith agent updates his location in order to increase his payoff. If the current locations of the players are denoted by $x_1(t), x_2(t), \ldots, x_n(t)$, the position of agent i at the next time step would be

$$\begin{aligned} x_i(t+1) &= \operatorname*{argmin}_x \sum_{j=1}^{n} \min\{\|x - x_j(t)\|^2, \epsilon^2\} \\ &= \operatorname*{argmin}_x \sum_{j\in\mathcal{N}_i(t)} \|x - x_j(t)\|^2 = \frac{\sum_{j\in\mathcal{N}_i(t)} x_j(t)}{|\mathcal{N}_i(t)|}. \end{aligned}$$

That establishes the equivalence between the better response dynamics and the updating process in the asynchronous Hegselmann-Krause model. □

Proposition 5.3 *An action profile $(x_1^*, x_2^*, \ldots, x_n^*)$ is a Nash equilibrium of the network formation game if and only if it is a steady state of the asynchronous Hegselmann-Krause dynamics.*

Proof Given an arbitrary Nash equilibrium $(x_1^*, x_2^*, \dots, x_n^*)$, we show that it is a steady state of the asynchronous Hegselmann-Krause dynamics by showing that for all $i, j \in [n]$ we either have $x_i^* = x_j^*$, or $\|x_i^* - x_j^*\| > \epsilon$. To show this by contradiction, let us assume that there are two players at locations $x_p^* \neq x_q^*$ such that $\|x_p^* - x_q^*\| \leq \epsilon$. Let $L = \{x_p^*, x_q^*, x_{\ell_1}^*, \dots, x_{\ell_s}^*\}$ denote the set of all the players' actions at this equilibrium point which are in the same connected component as x_p^* and x_q^* in the communication graph. Denoting one of the extreme points of $conv(L)$ by x_ℓ^* and using Lemma 5.2, it is not hard to see that player ℓ's action is not his best response, i.e., $\frac{\sum_{j\in\mathcal{N}_\ell^*} x_j^*}{|\mathcal{N}_\ell^*|} \neq x_\ell^*$, where $\mathcal{N}_\ell^* = \{j : \|x_j^* - x_\ell^*\| \leq \epsilon\}$. This is in contrast with the assumption of $(x_1^*, x_2^*, \dots, x_n^*)$ being a Nash equilibrium. To show that every steady state of the asynchronous Hegselmann-Krause dynamics is a Nash equilibrium of the network formation game is quite straightforward. □

Next we show that the above network formation game is, indeed, a potential game, with the sum of the utilities as a potential function. A further result (Corollary 5.1) shows directly that it is strategically equivalent to a team problem.

Theorem 5.3 *The network formation game is an ordinal potential game with a potential function of $U(x_1, x_2, \dots, x_n) = \sum_{i=1}^n U_i(x_i, x_{-i})$. In particular, we have*

$$U(x_i, x_{-i}) - U(x_i', x_{-i}) \leq -2|\mathcal{N}_i| \|x_i - x_i'\|^2,$$

where x_i' denotes the deviation of the ith player from action x_i to his better response $x_i' = \frac{1}{|\mathcal{N}_i|}\sum_{j\in\mathcal{N}_i} x_j$, and x_{-i} denotes the actions of all players except the ith one.

Proof Let $\mathcal{N}_i$ and $\mathcal{N}_i'$ denote the set of neighbors of player i before and after deviating, respectively. By definition of the payoff function of players (5.13), we can write

$$\begin{aligned} U(x_i, x_{-i}) - U(x_i', x_{-i}) &= \sum_{j\in\mathcal{N}_i\cup\mathcal{N}_i'} \big(U_j(x_i, x_{-i}) - U_j(x_i', x_{-i})\big) \\ &= U_i(x_i, x_{-i}) - U_i(x_i', x_{-i}) \\ &\quad + \sum_{j\in\mathcal{N}_i\cap\mathcal{N}_i'} \big(U_j(x_i, x_{-i}) - U_j(x_i', x_{-i})\big) \\ &\quad + \sum_{j\in\mathcal{N}_i\setminus\mathcal{N}_i\cap\mathcal{N}_i'} \big(U_j(x_i, x_{-i}) - U_j(x_i', x_{-i})\big) \\ &\quad + \sum_{j\in\mathcal{N}_i'\setminus\mathcal{N}_i\cap\mathcal{N}_i'} \big(U_j(x_i, x_{-i}) - U_j(x_i', x_{-i})\big), \quad (5.14) \end{aligned}$$

where the first equality is due to the fact that the utility of the players who do not observe x_i or x_i' does not change.

Next, we compute each of the summands in the above expression. Note that only the action of player i changes from x_i to x_i', while all others' actions remain unchanged (Fig. 5.3). We can write

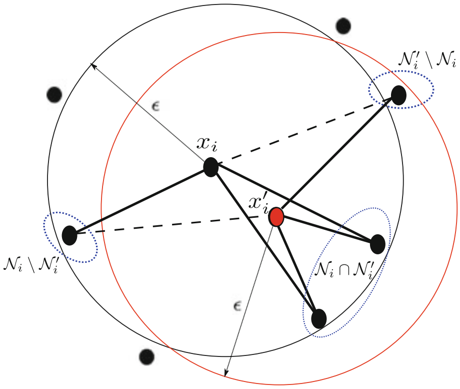

$\mathcal{N}_i' \setminus \mathcal{N}_i$
ϵ
x_i
x_i'
$\mathcal{N}_i \setminus \mathcal{N}_i'$
$\mathcal{N}_i \cap \mathcal{N}_i'$
ϵ

Fig. 5.3 Deviation of the ith player by updating to his better response x_i'

$$
\begin{aligned}
U_j(x_i, x_{-i}) - U_j(x_i', x_{-i}) &= \|x_j - x_i'\|^2 - \|x_j - x_i\|^2, \quad j \in \mathcal{N}_i \cap \mathcal{N}_i' \\
U_j(x_i, x_{-i}) - U_j(x_i', x_{-i}) &= \epsilon^2 - \|x_j - x_i\|^2, \qquad\qquad j \in \mathcal{N}_i \setminus \mathcal{N}_i \cap \mathcal{N}_i' \\
U_j(x_i, x_{-i}) - U_j(x_i', x_{-i}) &= \|x_j - x_i'\|^2 - \epsilon^2, \qquad\qquad j \in \mathcal{N}_i' \setminus \mathcal{N}_i \cap \mathcal{N}_i'.
\end{aligned} \tag{5.15}
$$

The reason for the first equality in (5.15) is that after the ith player deviates, every agent in $j \in \mathcal{N}_i \cap \mathcal{N}_i'$ still holds his connection with i, and hence, by the definition of the payoff function (5.13), his payoff is subjected to a change of $\|x_j - x_i'\|^2 - \|x_j - x_i\|^2$. (Note that all players except the ith one are kept fixed.) Similarly, every player $j \in \mathcal{N}_i \setminus \mathcal{N}_i \cap \mathcal{N}_i'$ stays connected to x_i while disconnecting his link with the ith player after i's deviation (since agent i gets far from him by moving from x_i to x_i', and hence they both prefer to stop building the road and each pay \$$\epsilon^2$ to the government). Therefore, the amount of change in the jth player's payoff is equal to $\epsilon^2 - \|x_j - x_i\|^2$. In a similar way, one can observe that the third equality in (5.15) holds. By the same line of argument and because of symmetry, one can easily show that the amount of change in the ith player's payoff is equal to the sum of all the terms in (5.15) over $j \in \mathcal{N}_i \cup \mathcal{N}_i'$. In fact, we can write

$$
\begin{aligned}
U_i(x_i, x_{-i}) - U_i(x_i', x_{-i}) = &\sum_{j \in \mathcal{N}_i \cap \mathcal{N}_i'} \left(\|x_j - x_i'\|^2 - \|x_j - x_i\|^2 \right) \\
&+ \sum_{j \in \mathcal{N}_i \setminus \mathcal{N}_i \cap \mathcal{N}_i'} (\epsilon^2 - \|x_j - x_i\|^2) + \sum_{j \in \mathcal{N}_i' \setminus \mathcal{N}_i \cap \mathcal{N}_i'} (\|x_j - x_i'\|^2 - \epsilon^2) \\
= &(|\mathcal{N}_i| - (|\mathcal{N}_i \cap \mathcal{N}_i'|))\epsilon^2 - (|\mathcal{N}_i'| - (|\mathcal{N}_i \cap \mathcal{N}_i'|))\epsilon^2 \\
&+ \sum_{j \in \mathcal{N}_i'} \|x_j - x_i'\|^2 - \sum_{j \in \mathcal{N}_i} \|x_j - x_i\|^2 \\
\leq &\sum_{j \in \mathcal{N}_i \setminus \mathcal{N}_i \cap \mathcal{N}_i'} \|x_j - x_i'\|^2 - \sum_{j \in \mathcal{N}_i' \setminus \mathcal{N}_i \cap \mathcal{N}_i'} \|x_j - x_i'\|^2 + \sum_{j \in \mathcal{N}_i'} \|x_j - x_i'\|^2 \\
&- \sum_{j \in \mathcal{N}_i} \|x_j - x_i\|^2 \\
= &\sum_{j \in \mathcal{N}_i} \|x_j - x_i'\|^2 - \sum_{j \in \mathcal{N}_i} \|x_j - x_i\|^2,
\end{aligned} \tag{5.16}
$$

where in the last inequality we have used the facts that

$$(|\mathcal{N}_i| - (|\mathcal{N}_i \cap \mathcal{N}_i'|))\epsilon^2 \leq \sum_{j \in \mathcal{N}_i \setminus \mathcal{N}_i \cap \mathcal{N}_i'} \|x_j - x_i'\|^2$$
$$(|\mathcal{N}_i'| - (|\mathcal{N}_i \cap \mathcal{N}_i'|))\epsilon^2 \geq \sum_{j \in \mathcal{N}_i' \setminus \mathcal{N}_i \cap \mathcal{N}_i'} \|x_j - x_i'\|^2.$$

(Note that $\|x_j - x_i'\|^2 \geq \epsilon^2$, if $j \in \mathcal{N}_i \setminus \mathcal{N}_i \cap \mathcal{N}_i'$, and $\|x_j - x_i'\|^2 \leq \epsilon^2$, if $j \in \mathcal{N}_i' \setminus \mathcal{N}_i \cap \mathcal{N}_i'$.) Substituting (5.15) and (5.16) in (5.14) and using (5.16), we get

$$\begin{aligned} U(x_i, x_{-i}) - U(x_i', x_{-i}) &\leq 2\left[\sum_{j \in \mathcal{N}_i} \|x_j - x_i'\|^2 - \sum_{j \in \mathcal{N}_i} \|x_j - x_i\|^2\right] \\ &= -2|\mathcal{N}_i|\|x_i - x_i'\|^2, \end{aligned}$$

where the last equality comes from substituting $x_i' = \frac{1}{\mathcal{N}_i}\sum_{j \in \mathcal{N}_i} x_j$ because player i deviates to his better place (Lemma 5.2). □

Corollary 5.1 *The network formation game is strategically equivalent to a team problem.*

Proof For any arbitrary player $i \in [n]$, let us define:

$$\beta(x_{-i}) = (n-1)(n-2)\epsilon^2 - \sum_{r,s \in [n] \setminus \{i\}} \min\{\|x_r - x_s\|^2, \epsilon^2\}.$$

Note that $\beta(x_{-i})$ depends on the actions of all the players except the ith player. By definition of $U(x_1, \ldots, x_n) = \sum_{k=1}^{n} U_k(x_k, x_{-k})$, we can write

$$2U_i(x_i, x_{-i}) + \beta(x_{-i}) = U(x_1, x_2, \ldots, x_n).$$

This shows that the network formation game is essentially a team problem, in the sense that every Nash equilibrium of the game is a person-by-person optimal solution for the team, and vice versa. More details on such strategic equivalence can be found in [16]. □

Now we are ready to provide an upper bound on the expected number of steps until the asynchronous Hegselmann-Krause dynamics with a uniform updating scheme reach their δ-equilibrium.

Theorem 5.4 *The expected number of steps until the agents in the asynchronous Hegselmann-Krause dynamics with a uniform updating schedule reach a δ-equilibrium is bounded from above by $2n^9(\frac{\epsilon}{\delta})^2$.*

Proof We evaluate the expected increase of the potential function given in Theorem 5.3. Since each player is chosen independently and with probability $\frac{1}{n}$, we have

$$\begin{aligned}\mathbb{E}[U(t+1)-U(t)] &= \sum_{i=1}^{n}\frac{1}{n}\mathbb{E}[U(t+1)-U(t)|i \text{ updates}]\\ &\geq 2\sum_{i=1}^{n}\frac{|\mathcal{N}_i(t)|}{n}\|x_i(t)-x_i(t+1)\|^2\\ &\geq \frac{2}{n}\sum_{i=1}^{n}\|x_i(t)-x_i(t+1)\|^2, \qquad (5.17)\end{aligned}$$

where in the first inequality we have used the result of Theorem 5.3.

Now using the result of Theorem 5.1 and by the same argument as in derivation of (5.5), we know that as long as there is a nontrivial δ component, we must have $\sum_{k=1}^{d}\bar{c}_k \geq \frac{\delta^2}{4}$, and therefore, $\sum_{i=1}^{n}\|x_i(t)-x_i(t+1)\|^2 \geq \frac{\delta^2}{n^6}$. Moreover, since $U(\tau) < n^2\epsilon^2$, we conclude that the expected number of times that nontrivial components of a diameter larger than $\delta > 0$ will emerge is bounded from above by $2n^9(\frac{\epsilon}{\delta})^2$. □

In fact, in the case of scalar asynchronous Hegselmann-Krause dynamics, one could come up with a sharper bound which we state in the following theorem.

Theorem 5.5 *The expected number of steps until the scalar asynchronous Hegselmann-Krause dynamics reach an $\frac{\epsilon}{n}$-equilibrium is bounded from above by $n^{5+2\log_n(n+1)}+n$.*

Proof Consider a particular time instant t, and let $x_1(t)=\min_{k\in[n]}x_k(t)$ and $x_m(t)=\max_{k\in\mathcal{N}_1(t)}x_k(t)$. Also, without loss of generality, let us assume that $x_1(t)=0$. It is clear that if $x_m(t) > \frac{\epsilon}{n^\alpha}$ and agent 1 updates, then we will have $x_1(t+1) > \frac{\epsilon}{n^{1+\alpha}}$, where α is a number to be determined later. In this case, the expected potential function will increase by at least $\frac{2}{n}\|x_1(t)-x_1(t+1)\|^2 \geq \frac{2\epsilon^2}{n^{3+2\alpha}}$. Otherwise, there is no other agent in the interval $[\frac{\epsilon}{n^\alpha},\epsilon]$. Now we consider two cases (Fig. 5.4):

- Agent $x_m(t)$ has a neighbor in the interval $(\epsilon, x_m(t)+\epsilon]$. Assuming that agent m updates, we will have $x_m(t+1)\geq\frac{x_m(t)+\epsilon}{n}$, and hence,

$$\|x_m(t+1)-x_m(t)\|^2 \geq \|\frac{\epsilon}{n}-x_m(t)\|^2 \geq (\frac{\epsilon}{n}-\frac{\epsilon}{n^\alpha})^2.$$

  Therefore, in this case and using (5.17), the amount of increase in the expected potential function is at least $\frac{2\epsilon^2}{n^3}(1-\frac{1}{n^{\alpha-1}})^2$.
- Agent $x_m(t)$ does not have any neighbor in the interval $(\epsilon, x_m(t)+\epsilon]$. We note that all the agents in the interval $[0, x_m(t)]$ form a cluster that is separated from other agents by a distance of at least ϵ. Noting that two separate clusters of nodes on a real line will stay apart from each other in the rest of the dynamics, we can decompose the original dynamics into two groups and analyze each of them separately.

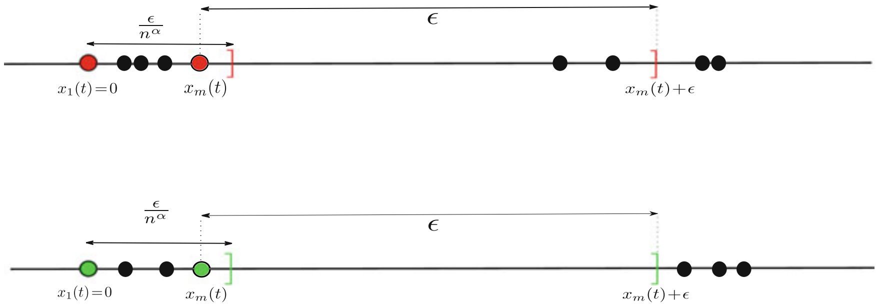


Fig. 5.4 Illustration of two different cases in the proof of Theorem 5.5

By choosing $\alpha = \log_n(n+1)$, we get $\frac{2\epsilon^2}{n^{3+2\alpha}} = \frac{2\epsilon^2}{n^3}(1 - \frac{1}{n^{\alpha-1}})^2$, and we can see that either we have an increase of size $\frac{2\epsilon^2}{n^{3+2\log_n(n+1)}}$ in the expected potential function, or the dynamics decompose into a cluster of size at most $\frac{\epsilon}{n^\alpha} < \frac{\epsilon}{n}$ and another part. Since the expected potential function cannot increase more than $n^{5+2\log_n(n+1)}$ number of steps ($U(\cdot) \leq n^2\epsilon^2$) and we cannot have more than n clustering decomposition, the expected number of steps until the dynamics decompose to clusters whose size is at most $\frac{\epsilon}{n}$ is bounded from above by $n^{5+2\log_n(n+1)} + n$. □

Remark 11 From the above lemma, after the expected number of $n^{5+2\log_n(n+1)} + n \approx n^7$, every agent lies within a cluster of diameter at most $\frac{\epsilon}{n}$, and those always are separated from each other by a distance of at least ϵ. Therefore, each agent in a cluster can observe the others, and henceforth, the diameter of the convex hull of each of the clusters shrinks very fast.

In the following, we provide a bound on the expected number of switching topologies during the evolution of the asynchronous Hegselmann-Krause process.

Theorem 5.6 *The expected number of switching topologies of the asynchronous Hegselmann-Krause dynamics with a uniform updating scheme is bounded from above by* $16n^9$.

Proof We show that switching topologies substantially increase the expected value of the potential function. To see that, first assume that the opinion profile at time $t-1$, i.e., $x(t-1)$, is $\frac{\epsilon}{2}$-trivial, and that updating some agent i at this time causes a switch in the topology of the network. We claim that the next profile, i.e., $x(t)$, is not $\frac{\epsilon}{2}$-trivial. Note that since there is a switch at time t and that within each of the $\frac{\epsilon}{2}$-trivial components each agent is able to observe the others, the convex hull of such a component shrinks even further after the updating of any agent in the component. Therefore, the switches must occur between the $\frac{\epsilon}{2}$-trivial components and not within them.

Now, let us assume that $i \in C_p$ (C_p denotes an $\frac{\epsilon}{2}$-trivial component) and that updating agent i at time $t-1$ makes him visible to another agent j in a different $\frac{\epsilon}{2}$-trivial component C_q (Fig. 5.5). Since C_p is an $\frac{\epsilon}{2}$-trivial component and the agents

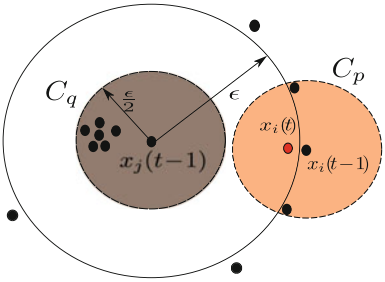


Fig. 5.5 Switching topology at time t from an $\frac{\epsilon}{2}$-trivial profile $x(t-1)$

in C_p are all the agents who are visible to agent i at time $t-1$, the movement of agent i from $x_i(t-1)$ to $x_i(t)$ can be at most $\frac{\epsilon}{2}$. Moreover, since agents j and i belong to different $\frac{\epsilon}{2}$-trivial components, their distance at time $t-1$ was larger than ϵ. That means that such a switching causes i and j to make a link with a distance of at least $\frac{\epsilon}{2}$ in the profile $x(t)$.

Now we partition all the possible switching times based on the profile at the previous time instant:

- Time t is a switching time, and $x(t-1)$ is an $\frac{\epsilon}{2}$-trivial profile. In this case and based on the above argument, $x(t)$ is not an $\frac{\epsilon}{2}$-trivial profile, and using the same argument as in relation (5.5) and in view of (5.9) and (5.12), we get $\sum_{k=1}^{n}\|x_k(t)-x_k(t+1)\|^2 \geq \frac{\epsilon^2}{16n^6}$.
- Time t is a switching time, and $x(t-1)$ is not an $\frac{\epsilon}{2}$-trivial profile. In this case and within a non-$\frac{\epsilon}{2}$-trivial component, using the same argument as in the first case, we get $\sum_{k=1}^{n}\|x_k(t-1)-x_k(t)\|^2 \geq \frac{\epsilon^2}{16n^6}$.

Therefore, if t is a switching time, using (5.17) we conclude that there is an increase of $\frac{\epsilon^2}{16n^6}$ at either time $t-1$ or t in the expected potential function. In other words, if t is a switching time, using (5.17) we can write

$$\mathbb{E}[U(t+1)-U(t-1)] = \mathbb{E}[U(t+1)-U(t)] + \mathbb{E}[U(t)-U(t-1)] \geq \frac{2}{n}\frac{\epsilon^2}{16n^6} = \frac{\epsilon^2}{8n^7}.$$

Now, given an arbitrary initial profile $x(0)$, let us use p_t to denote the probability of occurrence of a switching at time $t = 1, 2, \ldots$. Therefore, the amount of increase in the expected potential function is at least $\sum_{t=0}^{\infty} p_t \frac{\epsilon^2}{16n^7}$ (since we may count each instant twice). On the other hand, since $U(\tau) \leq n^2\epsilon^2, \forall \tau = 1, 2, \ldots$, we conclude that $\sum_{t=0}^{\infty} p_t$. But $\sum_{t=0}^{\infty} p_t$ is exactly equal to the expected number of switching topologies. Therefore, the expected number of switching topologies is bounded from above by $16n^9$. □

5.4 Heterogeneous Hegselmann-Krause Dynamics

Once again we consider the Hegselmann-Krause model (5.2), but this time we assume that each agent i has his or her own bound of confidence ϵ_i, which could be different from the others. Therefore, $\mathcal{N}_i(x(t)) = \{1 \leq j \leq n : \|x_i(t) - x_j(t)\| \leq \epsilon_i\}$ and $A(t), t \geq 0$ will change correspondingly. That causes an asymmetry for the interactions among the agents. In other words, there is a possibility that one agent $x_i(t)$ observes agent $x_j(t)$ but not vice versa. In fact, we are interested in studying the convergence behavior of such dynamics. In contrast with the homogeneous Hegselmann-Krause model, which reaches its steady state after finite time, the following example shows that in the heterogeneous case, steady state may not be reached in finite time.

Example 5.1 Consider three agents x_1, x_2, x_3 that are located at $-1, \frac{1}{3}, 1$, respectively, at the initial time $t = 0$. Also, let us assume $\epsilon_1 = \frac{1}{2}, \epsilon_2 = 2, \epsilon_3 = \frac{1}{2}$. As can be seen, agent x_2 is able to see all the agents at each time step. Therefore, after the first iteration, $x_2(1) = \frac{-1+\frac{1}{3}+1}{3} = \frac{1}{3^2}$, and since the confidence bounds of x_1 and x_3 are small, they can see no one except themselves, and hence they will remain in their own locations. Therefore, at time $t = 1$, we will have $x_1(1) = -1, x_2(1) = \frac{1}{3^2}, x_3(1) = 1$. With the same line of argument, it is not hard to see that at any time instant $t = 1, 2, \ldots$ the position of agents will be $x_1(t) = -1, x_2(t) = \frac{1}{3^{t+1}}, x_3(t) = 1$. That shows that the dynamics will converge to their steady state $(-1, 0, 1)$, but not in finite time.

In the above example, one of the main reasons that the convergence was not achieved in finite time was that there were two agents who did not have interaction with others in the dynamics and remained fixed without any movement forever. We refer to such agents as *silent agents*. In the next theorem, we show that if the amount of time an agent sleeps (is inactive) is finite, then we will have finite time convergence of the dynamics to their steady state. We note that similar types of such asynchronous analysis under different scenarios and settings can be found in [17–19].

Theorem 5.7 *Consider the heterogeneous Hegselmann-Krause model, where the ith agent $i \in [n]$ has a confidence bound of $\epsilon_i > 0$. Also, assume that there is an integer T^* such that no agent is silent for a period of time longer than T^*. Then, the dynamics will converge to their steady state in finite time.*

Proof We prove the theorem by induction on the number of agents. For $n = 1$ the result is obvious, and the initial time is the termination time. Let us assume that the result holds for each $k \leq n$, and now suppose that we have $n+1$ agents with different confidence bounds. We show that there is a finite time T such that the left product of every T consecutive matrices $A(t), t \geq 0$ of the dynamics will generate a matrix with at least one positive column.

Starting from agent 1, let us define

$$S(t) = \{i \in [n+1] | (A(t)A(t-1)\ldots A(0))_{i1} > 0\},$$

and $S^c(t) = [n+1]\backslash S(t)$ to be its complement. Since each agent can see itself at each time instant, if $i \in S(t)$ for some time t, then it will be in $S(t')$ for all $t' \geq t$. In other words, we have $S(0) \subseteq S(1) \subseteq S(2) \subseteq \ldots$. Now we claim that there must be a finite time T such that $S(T) = [n+1]$. Otherwise, let us assume that there exists a time instant t_0 such that $S(t_0) = S(t), \forall t > t_0$. By the definition of $S(t)$, that means that for $t > t_0$, none of the agents in $S^c(t)$ can see any agent in $S(t)$ (although it may happen that some agents in $S(t)$ are still able to see some of the agents in $S^c(t)$). That means that the agents in the set $S^c(t_0)$ constitute a group of agents whose opinions in the future of the dynamics $t \geq t_0$ will not be influenced by any other agent in $S(t_0)$. On the other hand, since $|S^c(t_0)| \leq n$ (note that $S(0) = \{1\}$), according to the induction assumption, the agents in $S^c(t_0)$ will reach their steady state after some finite time T_n, where T_n denotes the maximum number of steps for n agents to reach their steady state, which, by induction assumption, is considered to be a finite number. However, under the hypothesis of the Theorem, after reaching the steady state, these agents cannot remain silent for more than T^* more steps. Therefore, after a finite time $T^* + T_n$, at least one more agent will be added to the set $S(t_0)$, and the cardinality of $S(t_0)$ will increase by at least 1. Since the total number of agents is $n+1$, $T := (n+1)(T^* + T_n)$ steps are enough to guarantee $S(T) = [n+1]$. That shows that $A(T)\ldots A(1)A(0)$ will be a matrix in which the first column will be strictly positive.

On the other hand, since all the positive entries of those matrices are bounded from below by $\min^+(A(t)) \geq \frac{1}{n+1}$, the minimum positive entry of the left product of every T consecutive such matrices will be larger than $(\frac{1}{n+1})^T$. Using Lemma 2.9, we can see that after every T steps, the diameter of the convex hull of the agents' opinions will shrink by a factor of at least $1 - (\frac{1}{n+1})^T$. Therefore, there exists a finite time $T_{n+1} < \infty$ such that the diameter of the convex hull of the agents' opinions at time T_{n+1} is smaller than $\min_{i \in [n+1]} \epsilon_i$. That means that after T_{n+1} steps, every agent is able to observe the others in his or her own neighborhood, and in the next step, the dynamics reach a steady state. □

In fact, the above theorem asserts that if there exists an external input which creates an incentive for the agents to interact with someone else after some period of time, then the circulation of information in the society will be sufficient to guarantee the finite time formation of the opinions.

5.5 Discussion

Inspired by the results given in Sect. 5.3, we will now discuss some of the possible directions that could be pursued to analyze the asynchronous heterogeneous Hegselmann-Krause model in more detail. In fact, because of the different confidence bounds, the symmetry from which we benefit in the homogeneous case no longer holds. Therefore, the communication topology in this case can be interpreted as a digraph (directed graph) instead of an undirected graph. In this case one way of showing the asymptotic convergence of the heterogeneous Hegselmann-Krause dynamics to a steady state is to design a proper utility function for each player such that the resulting network formation game changes to a team problem, such that each player's update contributes an increase (decrease) to a global function toward an equilibrium.

A natural idea here is to define the utilities of the players based on functions of their own confidence bounds and their relative distances from others such that their better response dynamics coincide with the evolution of the asynchronous heterogeneous Hegselmann-Krause dynamics. For example, one may define the utility of the ith player to be $U_i(t) = (n-1)\epsilon_i^2 - \sum_{j=1}^{n} \min\{(x_i(t) - x_j(t))^2, \epsilon_i^2\}$, where ϵ_i denotes the confidence bound of the ith agent and $x(t) = (x_1(t), x_2(t), \ldots, x_n(t))$ denotes the opinion profile at time instant t. It turns out that such utility functions do not make the network formation game a potential game or lead to a strategically equivalent team problem. However, one can consider 15 different possibilities for creation or breaking of edges among agents, assuming that only one agent updates (deviates) to a new position. In that case, one can think of a proper weighting on the edges in order to *distinguish* one-sided edges from symmetric (two-sided) edges. For example, if there is a one-sided edge from player i to player j, one can rescale the utility of agent i by a fraction of his own confidence bound and his neighbors' in order to adjust the influence of other players' actions on his own utility function. At this point, we are not aware of any such utility functions, and we leave the full analysis of the heterogeneous Hegselmann-Krause dynamics as a direction for future research.

5.6 Conclusion

In this chapter, we studied the termination time of the Hegselmann-Krause dynamics in finite dimensions and under various settings: synchronous, asynchronous, homogeneous, and heterogeneous. We provided a polynomial upper bound for the termination time of the synchronous homogeneous model independent of the dimension of the ambient space. We showed that the asynchronous Hegselmann-Krause model can be formulated as a sequence of better response dynamics of an ordinal potential game. Furthermore, we provided an upper bound for the expected number of steps until the dynamics reach their δ-equilibrium. In particular, we bounded the expected number of switchings in the topology of the networks during the evolution of the system. We considered the heterogeneous Hegselmann-Krause dynamics, and we obtained a necessary condition for finite time convergence of such dynamics. Finally, we discussed some of the possible future directions that could be pursued to enable analysis of heterogeneous Hegselmann-Krause dynamics in more detail.

References

1. R. Hegselmann and U. Krause, "Opinion dynamics and bounded confidence models, analysis, and simulation," *Artificial Societies and Social Simulation*, vol. 5, pp. 1–33, 2002.
2. F. Bullo, J. Cortes, and S. Martinez, *Distributed Control of Robotic Networks*. Princeton University Press, 2009.
3. Y. Dong, X. Chen, H. Liang, and C.-C. Li, "Dynamics of linguistic opinion formation in bounded confidence model," *Information Fusion*, vol. 32, pp. 52–61, 2016.
4. H.-T. Wai, A. Scaglione, and A. Leshem, "Identifying trust in social networks with stubborn agents, with application to market decisions," in *53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE, 2015, pp. 747–754.
5. M. Pineda, R. Toral, and E. Hernández-García, "The noisy Hegselmann-Krause model for opinion dynamics," *The European Physical Journal B*, vol. 86, no. 12, pp. 1–10, 2013.
6. S. Kurz and J. Rambau, "On the Hegselmann-Krause conjecture in opinion dynamics," *Journal of Difference Equations and Applications*, vol. 17, no. 6, pp. 859–876, 2011.
7. J. M. Hendrickx and A. Olshevsky, "On symmetric continuum opinion dynamics," *Proc. 52nd IEEE Conference on Decision and Control (CDC)*, pp. 1989–1994, 2013.
8. V. D. Blondel, J. M. Hendrickx, and J. N. Tsitsiklis, "On the 2r conjecture for multi-agent systems," in *Proceedings of the European Control Conference 2007 (ECC1707).* Citeseer, 2007, pp. 874–881.
9. J. Lorenz, "Continuous opinion dynamics under bounded confidence: A survey," *International Journal of Modern Physics C*, vol. 18, no. 12, pp. 1819–1838, 2007.
10. S. R. Etesami and T. Başar, "Game-theoretic analysis of the Hegselmann-Krause model for opinion dynamics in finite dimensions," *IEEE Transactions on Automatic Control, 60(7):1886–1897, July 2015.*
11. B. Touri and A. Nedić, *On existence of a quadratic comparison function for random weighted averaging dynamics and its implications*. Proceedings of the 50th IEEE Conference on Decision and Control and European Control Conference, CDC-ECC 2011, Orlando, FL, USA, December 12–15, 2011, 2009.
12. M. Roozbehani, A. Megretski, and E. Frazzoli, "Lyapunov analysis of quadratically symmetric neighborhood consensus algorithms," in *Proc. 47th IEEE Conference on Decision and Control (CDC)*. IEEE, 2008, pp. 2252–2257.
13. S. R. Etesami, T. Başar, A. Nedić, and B. Touri, "Termination time of multidimensional Hegselmann-Krause opinion dynamics," in *Proc. American Control Conference (ACC)*. IEEE, 2013, pp. 1255–1260.
14. A. Bhattacharyya, M. Braverman, B. Chazelle, and H. L. Nguyen, "On the convergence of the Hegselmann-Krause system," in *Proceedings of the 4th Conference on Innovations in Theoretical Computer Science*. ACM, 2013, pp. 61–66.
15. B. Chazelle, "The total s-energy of a multiagent system," *SIAM Journal on Control and Optimization*, vol. 49, no. 4, pp. 1680–1706, 2011.
16. T. Başar and G. J. Olsder, *Dynamic Noncooperative Game Theory*. SIAM, 1999, vol. 23.
17. S. Li and T. Başar, "Asymptotic agreement and convergence of asynchronous stochastic algorithms," *IEEE Transactions on Automatic Control*, vol. 32, no. 7, pp. 612–618, 1987.
18. D. P. Bertsekas, "Distributed asynchronous computation of fixed points," *Mathematical Programming*, vol. 27, no. 1, pp. 107–120, 1983.
19. J. N. Tsitsiklis and M. Athans, "Convergence and asymptotic agreement in distributed decision problems," *IEEE Transactions on Automatic Control*, vol. 29, no. 1, pp. 42–50, 1984.

Chapter 6
Capacitated Selfish Replication Game

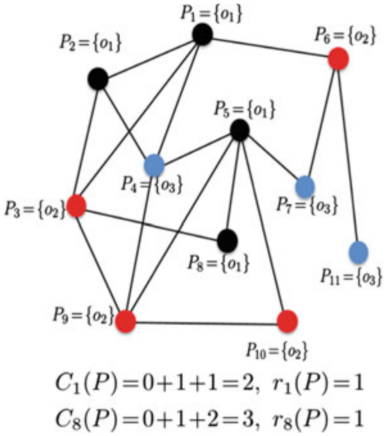
In this chapter we introduce the class of capacitated selfish replication games as was introduced in [1]; for simplicity we will refer to them as CSR games. Typically, CSR games are defined in terms of a set of available resources for each player, where the players are allowed to communicate through an undirected communication graph. Such a communication graph identifies the access cost among the players, and the goal for each player is to satisfy his/her customers' needs with minimum cost. Ideally, and in order to avoid any additional cost, each player only wants to use his/her own set of resources. However, due to capacity limitations, the players cannot store and have immediate access to all the resources they may need, and hence, they will have to search in the network and borrow some of the resources which are not available in their own caches from others in order to meet their customers' demands, and such a transaction incurs cost. In the following we first provide a formal definition of CSR games.

6.1 CSR Game Model

We start with a set of $[n] = \{1, 2, \ldots, n\}$ nodes (players) which are connected by an undirected graph $\mathcal{G} = ([n], \mathcal{E})$. We denote the set of all resources by $O = \{o_1, o_2, \ldots, o_k\}$. For simplicity, but without much loss of generality, we assume that each node can hold only one resource in its cache. All the results can in fact be extended quite readily to CSR games with different capacities (see later Remark 12). Moreover, we assume that each node has access to all the resources. For a particular allocation $P = (P_1, P_2, \ldots, P_n)$, we define the sum cost function $C_i(P)$ of the ith player as follows:

S.R. Etesami, *Potential-Based Analysis of Social, Communication, and Distributed Networks*, Springer Theses, DOI 10.1007/978-3-319-54289-8_6

Fig. 6.1 CSR game with $n = 11$ players and $O = \{o_1, o_2, o_3\}$ resources



$$C_i(P) = \sum_{o \in O \setminus \{P_i\}} d_{\mathcal{G}}(i, \sigma_i(P, o)), \tag{6.1}$$

where $\sigma_i(P, o)$ is i's nearest node holding o in P. Further, given an allocation profile P, we define the ***radius*** of agent i, denoted by $r_i(P)$, to be the distance between node i and the nearest node other than her holding the same resource as i, i.e., $r_i(P) = \min_{j \neq i, P_j = P_i} d_{\mathcal{G}}(i, j)$. If there does not exist such a node, we simply define $r_i(P) = D$, where D is the diameter of the network. We suppress the dependence of $r_i(P)$ on P whenever there is no ambiguity. In Fig. 6.1 we have illustrated the CSR game for $n = 11$ players and $|O| = 3$ resources. In particular, the associated costs and radii for two players $i = 1, 8$ for some allocation profile P have been given. Finally, if some resource o is missing in an allocation profile P, we define the cost of each player for that specific resource to be $d_G(i, \sigma_i(P, o)) = D + 1, \forall i \in [n]$. Therefore, for $n \geq |O|$, this incentivizes at least one of the players to allocate the missing resources in the network. In the case where $n < |O|$, all the players will allocate different resources and the game becomes trivial; hence, we can simply assume that $n \geq |O|$.

Remark 12 Actually all the results and proofs in this chapter can be carried over to games with varying capacities. This can be done by constructing a new network which transfers games with different cache sizes to one with unit size caches; see Sect. 6.11.

Remark 13 Given two allocation profiles P and $\tilde{P}$, which only differ in the ith coordinate, using (6.1) and the definition of radius, one can easily see that $C_i(P) - C_i(\tilde{P}) = r_i(\tilde{P}) - r_i(P)$. This establishes an equivalence between decrease in cost and increase in radius for player i, when the actions of the remaining players are fixed.

As an example of the above CSR game model, consider a shopping network composed of many retailers (players). The entire shopping network can be operated by a chain of stores such as Walmart, an online website such as Amazon, or individually by retailers. There could be many items (resources) which are bargained through this network. Customers frequently refer to the closest retailers in their geographical locations and place their orders. Since retailers have limited storage capacity, it is quite possible that they may not have the requested item in stock. However, in order to remain credible, the retailers promise their customers to have the requested item in stock within a few days, in which case they have to acquire that item from other retailers in cities closest to them. Assuming that the travel (shipping) costs are proportional to the number of hops (roads) between different cities,[1] each retailer will try to minimize the sum of its travel costs for different possible requests of its customers given by (6.1). Since retailers only care about minimizing their own travel costs while meeting the requests of their own customers, this defines a game among retailers in which case choosing what items to store in stock becomes very critical.

Definition 6.1 Given an n-player CSR game, an allocation profile $P^* = (P_1^*, P_2^*, \ldots, P_n^*)$ is said to constitute a Nash equilibrium (NE) if $C_i(P_i^*, P_{-i}^*) \le C_i(P_i, P_{-i}^*)$, $\forall i \in [n], \forall P_i$, where P_{-i}^* denotes the allocated resources of all the players except the ith one.

In this chapter, one of our goals is to study the convergence of some particular updating dynamics to NE of the CSR game. Toward that end, we let $P_i(t)$ be the allocated resource for node i at some generic time $t = 1, 2, \ldots$, and $P(t) = (P_1(t), P_2(t), \ldots, P_n(t))$ be the allocation vector at that time. We suppress the dependence on time t whenever there is no ambiguity. Note that there is an important difference between an equilibrium and the dynamics that lead to that equilibrium. In fact, at an equilibrium each node has only one resource such that the entire profile satisfies the inequalities in Definition 6.1. But in the process of finding an equilibrium, players frequently replace their resources with others from $O = \{o_1, \ldots, o_k\}$ until there is no incentive for any individual to deviate. Note that when a player swaps its resource by another one at some time t, it does not restrict the actions of the other players, nor the total set of resources accessible to others. Another useful way to think about this process is to consider resources as different colors such that each node chooses a color and incurs some cost depending on the colors of the other nodes. Therefore, at every stage one player changes its color in order to minimize its own cost, and this continues until no other player wants to deviate.

In our analysis of the dynamics, we assume that updates take place one at a time, and no two updates take place simultaneously. In fact, it is not hard to see that a simultaneous update may result in some cyclic behavior in the state of the game. As an example, given two players which are connected by an edge, and two possible resources $\{o_1, o_2\}$, let us assume that at the beginning both players have resource o_1 in their own cache. Then if we allow the players to update simultaneously, in the next

[1] Communication network can be regarded, in the context of this example, as transportation network between different cities.

time step both of them will choose o_2. Hence, such simultaneous updates will never reach to an NE, i.e., the situation where one node has resource o_1 and the other one has resource o_2.

6.2 Global Optimality Versus Pure Nash Equilibrium over Trees

In this section we study the connection between global optimal allocation and pure Nash equilibrium points of the CSR game when the underlying network is restricted to be a *tree*. In fact, for tree networks one can obtain an NE of the game in only linear time. However, this result no longer holds when the underlying network has a cycle. Hence, in the subsequent sections we will use alternative methods to find or approximate NE points of the game over general networks.

Definition 6.2 We say that an allocation profile $P^{\text{o}} = (P_1^{\text{o}}, \ldots, P_n^{\text{o}})$ is an optimal allocation for the CSR game if $P^{\text{o}} \in \operatorname{argmin}_{P\in\mathcal{P}} \sum_{i=1}^{n} C_i(P)$, where $\mathcal{P}$ denotes the collection of all the possible allocations.

Definition 6.3 Given an undirected network $\mathcal{G}$, and an arbitrary node i, we let the *vertex density radius* $\tau_{_\mathcal{G}}^i \in \mathbb{N}$ be such that $|B_{\mathcal{G}}(i, \tau_{_\mathcal{G}}^i - 1)| < |O| \le |B_{\mathcal{G}}(i, \tau_{_\mathcal{G}}^i)|$. In other words, $\tau_{_\mathcal{G}}^i$ is the smallest positive integer such that a graphical ball of radius $\tau_{_\mathcal{G}}^i$ around node i contains at least $|O|$ nodes. For simplicity we sometimes suppress the subscripts $\mathcal{G}$ from the notations.

The following theorem states that when the underlying network has a tree structure, then every globally optimal allocation is a NE for the CSR game.

Theorem 6.1 *Assume that the network in the CSR game is a tree T of n nodes. Then every optimal allocation P^o which minimizes the total cost, i.e., $P^o \in \operatorname{argmin}_{P\in\mathcal{P}}$ $\sum_{i=1}^{n} C_i(P)$ must be a NE. Furthermore, there is an algorithm which reaches an optimal allocation in only n steps.*

Proof Let us start from an arbitrary node of T as a root and label it by 1. We define ℓth level to be the set of all nodes in T which are at distance ℓ from node 1. At the ith step of the algorithm, we choose an agent at the highest level who has not been chosen before; we label it by i, and let her allocate one resource based on her best response with respect to all the agents who have updated before, i.e., $1, 2, \ldots, i-1$. Ties are broken arbitrarily.

Claim: At the end of the algorithm and for every i, all the resources in $B_{_T}(i, \tau_{_T}^i - 1)$ are different, while every resource appears at least once in $B_{_T}(i, \tau_{_T}^i)$.

Proof of the Claim: We establish the claim by induction on the number of nodes. For $n = |O|$ the claim trivially holds, since the algorithm returns a profile wherein every resource appears exactly once. Such a profile is clearly an optimal allocation which satisfies the claim. Assuming that the above claim holds for the outcome of the algorithm over all trees of n nodes, let T_{n+1} be a tree of $n+1$ nodes. Based on the updating scheme, node $n+1$ is the last node that acts, which immediately implies that $n+1$ must be a leaf in the last level (with respect to the reference node 1) of the tree T_{n+1}. Removing this node and its adjacent edge from T_{n+1}, the remaining part is a tree of n nodes $T_n = T_{n+1} \setminus \{n+1\}$. Now let us denote the profile obtained at the end of the algorithm over T_{n+1} by $(P_1, P_2, \dots, P_n, P_{n+1})$. Because of the updating rule, it is clear that the first n steps of the algorithm over T_{n+1} exactly coincide with those given that we were running the algorithm over T_n. Therefore, by the induction hypothesis $(P_1, P_2, \dots, P_n)$ is an optimal allocation for T_n such that for $i \in [n]$ all the resources in $B_{T_n}(i, \tau^i_{T_n} - 1)$ are different while every resource appears at least once in $B_{T_n}(i, \tau^i_{T_n})$.

Next we show that when node $n+1$ plays its best response P_{n+1} with respect to the allocated resources $(P_1, P_2, \dots, P_n)$ on T_n, the claim still holds. To see this, let us consider an arbitrary $i \in [n]$. Clearly if $d_{T_{n+1}}(i, n+1) > \tau^i_{T_n}$, then $B_{T_{n+1}}(i, \tau^i_{T_{n+1}}) = B_{T_n}(i, \tau^i_{T_n})$. Thus by induction hypothesis and independent of what resource node $n+1$ updates to, $B_{T_{n+1}}(i, \tau^i_{T_{n+1}})$ contains all the resources while the resources in $B_{T_{n+1}}(i, \tau^i_{T_{n+1}} - 1) = B_{T_n}(i, \tau^i_{T_n} - 1)$ are all different. Now we consider two cases:

Case I: If $d_{T_{n+1}}(i, n+1) \le \tau^i_{T_n} - 1$, then $B_{T_n}(i, \tau^i_{T_n} - 1)$ will be a subset of the closest $|O| - 1$ vertices to node $n+1$ except itself (Fig. 6.2, left side). Therefore, when node $n+1$ wants to update based on his best response, he will consider all the resources which have appeared in $B_{T_n}(i, \tau^i_{T_n} - 1)$, and by induction hypothesis we know that all such resources are different. Since $|B_{T_n}(i, \tau^i_{T_n} - 1)| \le |O| - 1$, the chosen resource by player $n+1$, i.e., P_{n+1} has not appeared among the resources in $B_{T_n}(i, \tau^i_{T_n} - 1)$. Now since $B_{T_{n+1}}(i, \tau^i_{T_{n+1}} - 1) \subseteq B_{T_n}(i, \tau^i_{T_n} - 1) \cup \{n+1\}$ and all the resources in $B_{T_n}(i, \tau^i_{T_n} - 1) \cup \{n+1\}$ are different, after player $n+1$ updates, still all the resources in $B_{T_{n+1}}(i, \tau^i_{T_{n+1}} - 1)$ will be different. Similarly, since $B_{T_{n+1}}(n+1, \tau^{n+1}_{T_{n+1}} - 1) \subseteq B_{T_n}(i, \tau^i_{T_n} - 1) \cup \{n+1\}$, all the resources in $B_{T_{n+1}}(n+1, \tau^{n+1}_{T_{n+1}} - 1)$ will be different. On the other hand, either $B_{T_{n+1}}(n+1, \tau^{n+1}_{T_{n+1}}) = B_{T_n}(i, \tau^i_{T_n} - 1) \cup \{n+1\}$, which in this case $B_{T_{n+1}}(n+1, \tau^{n+1}_{T_{n+1}})$ has all the resources exactly once (and in particular contains all the resources), or $B_{T_{n+1}}(n+1, \tau^{n+1}_{T_{n+1}}) = B_{T_n}(i, \tau^i_{T_n}) \cup \{n+1\}$ in which case $B_{T_{n+1}}(n+1, \tau^{n+1}_{T_{n+1}})$ must contain all the resources (since by induction hypothesis $B_{T_n}(i, \tau^i_{T_n})$ contains all the resources). Similarly for player i we either have $B_{T_{n+1}}(i, \tau^i_{T_{n+1}}) = B_{T_n}(i, \tau^i_{T_n} - 1) \cup \{n+1\}$ in which case $B_{T_{n+1}}(i, \tau^i_{T_{n+1}})$ has all the resources exactly once (and in particular contains all the resources), or $B_{T_{n+1}}(i, \tau^i_{T_{n+1}}) = B_{T_n}(i, \tau^i_{T_n}) \cup \{n+1\}$ in which case $B_{T_{n+1}}(i, \tau^i_{T_{n+1}})$ must contain all the resources. Therefore overall we have shown that in Case I when player $n+1$ updates, the conditions of the claim still hold for players i and $n+1$.

Case II: If $d_{T_{n+1}}(i, n+1) = \tau^i_{T_n}$, then clearly $\tau^i_{T_{n+1}} = \tau^i_{T_n} = d_{T_{n+1}}(i, n+1)$ (Fig. 6.2, right side). Therefore, independent of what resource node $n+1$ updates, since $n+1 \notin B_{T_n}(i, \tau^i_{T_n} - 1) = B_{T_{n+1}}(i, \tau^i_{T_{n+1}} - 1)$, by induction hypothesis one can

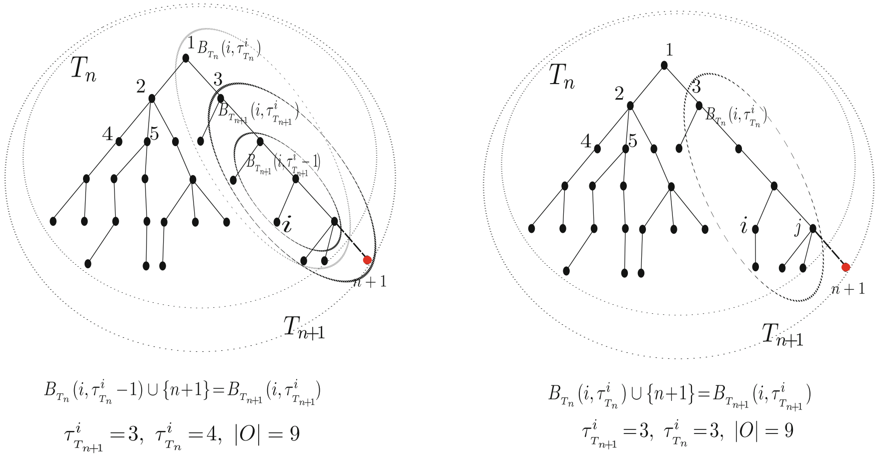

T_n
1
2
3
4
5
$B_{T_n}(i,\tau^i_{T_n})$
$B_{T_{n+1}}(i,\tau^i_{T_{n+1}})$
$B_{T_{n+1}}(i,\tau^i_{T_{n+1}}-1)$
i
$n+1$
T_{n+1}
$B_{T_n}(i,\tau^i_{T_n}-1)\cup\{n+1\}=B_{T_{n+1}}(i,\tau^i_{T_{n+1}})$
$\tau^i_{T_{n+1}}=3,\ \tau^i_{T_n}=4,\ |O|=9$
T_n
1
2
3
4
5
$B_{T_n}(i,\tau^i_{T_n})$
i
j
$n+1$
T_{n+1}
$B_{T_n}(i,\tau^i_{T_n})\cup\{n+1\}=B_{T_{n+1}}(i,\tau^i_{T_{n+1}})$
$\tau^i_{T_{n+1}}=3,\ \tau^i_{T_n}=3,\ |O|=9$

Fig. 6.2 Illustration of two different cases in the proof of Theorem 6.1

easily see that $B_{T_{n+1}}(i, \tau^i_{T_{n+1}} - 1)$ contains only different resources. Furthermore, $B_{T_n}(i, \tau^i_{T_n}) \cup \{n + 1\} = B_{T_{n+1}}(i, \tau^i_{T_{n+1}})$, and by induction we know that $B_{T_n}(i, \tau^i_{T_n})$ contains all the resources, which immediately implies that $B_{T_{n+1}}(i, \tau^i_{T_{n+1}})$ contains all the resources. On the other hand, to show that the claim conditions hold for player $n + 1$, let j be the closest node to $n + 1$. Using the induction hypothesis node j in T_n also satisfies the conditions of the claim. Now if $\tau^j_{T_n} \geq 2$, then since $1 = d_{T_{n+1}}(j, n + 1) \leq \tau^j_{T_n} - 1 = 1$, using Case I we conclude that $B_{T_{n+1}}(n + 1, \tau^{n+1}_{T_{n+1}} - 1)$ includes only different resources while $B_{T_{n+1}}(n + 1, \tau^{n+1}_{T_{n+1}})$ contains all the resources. But if $\tau^j_{T_n} = 1$, we have $\tau^{n+1}_{T_n} \leq 2$, and hence, the resource that node $n + 1$ updates to is different from j, i.e., $P_j \neq P_{n+1}$. Since $B_{T_{n+1}}(n + 1, \tau^{n+1}_{T_{n+1}} - 1) \subseteq \{P_j, P_{n+1}\}$, it only contains different resources, while $B_{T_{n+1}}(n + 1, \tau^{n+1}_{T_{n+1}})$ contains all the resources (note that all the resources appear at least once in $B_{T_n}(j, 1)$ and $n + 1$ is adjacent to node j). This again shows that in Case II when player $n + 1$ updates, the conditions of the claim still hold for players i and $n + 1$. This completes the induction process.

Therefore, we have shown that for any tree T, at the end of the algorithm all the resources in $B_T(i, \tau^i_T - 1)$ are different, while every resource appears at least once in $B_T(i, \tau^i_T)$. This means that at the end of the algorithm the cost of each player equals its minimum possible. This immediately implies that the final allocation profile has minimum total cost such that no player can decrease his own cost even further; hence, it must be a NE. □

6.3 Existence of Pure Nash Equilibrium over General Networks

As we saw in the previous section, when the underlying network is a tree, the CSR game has a pure Nash equilibrium which can be obtained in linear time. It turns out a pure Nash equilibrium always exists in the CSR game independent of the network structure. However, finding such an equilibrium in "efficient time" over general networks can be much more complicated, which is the subject of the rest of this chapter. Toward this end, we first state the following definition:

Definition 6.4 ([2, 3]) . A real valued function $\Phi(\cdot)$ over the finite set of all the allocation profiles is called a generalized ordinal potential function for the CSR game if for every player i and $P_i, P_i' \in O$, and any P_{-i}, we have

$$C_i(P_i, P_{-i}) - C_i(P_i', P_{-i}) > 0 \Rightarrow \Phi(P_i, P_{-i}) - \Phi(P_i', P_{-i}) > 0,$$

i.e., the changes in the costs are aligned with those of the potential function.

Here, we note that if a game admits a generalized ordinal potential function, it must have at least one NE [3]. This is simply because any minimizer of the generalized ordinal potential function gives a NE profile of actions in which no player can decrease its cost even further (as otherwise that profile is not a minimizer anymore). In particular, any sequence of the strictly better response moves[2] by the players converges to a minimizer of the potential function, and hence, a NE of the game. It is worth mentioning that potential games can be viewed to be strategically equivalent to team problems, in the sense that every Nash equilibrium of the game is a person-by-person optimal solution for the team, and vice versa. More details on such strategic equivalence can be found in [4].

Definition 6.5 Given a profile of allocations at time t, $P(t) = (P_1(t), \dots, P_n(t))$, on a graph with diameter D, we introduce a radius vector, $n(t) = (n_1(t), \dots, n_D(t))$, where $n_r(t),\ r = 1, \dots, D$, is the number of agents who have radius r with respect to the profile $P(t)$. Also, we let $N_r(t)$ be the set of all agents at time t whose radii are equal to r, that is, $|N_r(t)| = n_r(t)$.

It has been shown earlier in [1] that the CSR game admits a generalized ordinal potential function, and hence a NE. However, in the following we introduce a generalized ordinal potential function in a slightly different form, which will be more useful for our analysis.

Lemma 6.1 *In the CSR game, assume that at time step t an agent i whose radius is r updates its resource and strictly reduces its cost. Then there exists an integer $\alpha \geq 1$ such that $n(t+1)$ has the following form* $\Big(n_1(t), \dots, n_{r-1}(t), n_r(t) - \alpha, n_{r+1}$

[2] An action update for a player is based on a strictly better (best) response move if it strictly decreases (the most) the cost of that player.

$(t+1), \ldots, n_D(t+1)\Big)$. *In other words, the radius vector of the game will decrease lexicographically after each strictly better move of a player.*

Proof Let us assume that $P_i(t) = \{a\}$ and that player i wants to update its resource at time t to $P_i(t+1) = \{b\}$ at the next time step in order to strictly decrease its cost (and hence strictly increase its radius). Moreover, let us assume that $i \in N_r(t)$ for some $r \in \{1, \ldots, D-1\}$. After such an update, player i will leave the set $N_r(t)$ in favor of another set $N_{r'}(t+1)$ at the next time step by increasing its radius to some $r' > r$. We prove the lemma by showing

$$\begin{aligned}&N_k(t) = N_k(t+1), \quad \forall k \in \{1, \ldots, r-1\}\\&n_r(t+1) \leq n_r(t) - 1.\end{aligned} \tag{6.2}$$

To see this, let us consider another player $j \in N_k(t)$ for some arbitrary but fixed $k \in \{1, \ldots, r-1\}$, i.e., the radius of player j at time t, denoted by $r_j(t) = k$, is less than r. We consider three cases:

- $P_j(t) \notin \{a, b\}$: In this case updating i does not affect the radius of player j, i.e., player j will remain in its own radius set at the next time step. Thus $j \in N_k(t+1)$.
- $P_j(t) = \{a\}$: In this case since $r_j(t) = k < r$, the radius of j is determined by another player except i. In particular, updating player i's resource from $\{a\}$ to $\{b\}$ does not affect the radius of player j, i.e., player j will remain in its own set $j \in N_k(t+1)$.
- $P_j(t) = \{b\}$: In this case, one can again see that $r_j(t)$ is fully determined by a player other than i. Moreover, since after updating i from $P_i(t) = \{a\}$ to $P_i(t+1) = \{b\}$, agent i's radius will even increase further to r', we conclude that the graphical distance between j and i must be at least r'. Since we already know $r' > r > k$, this shows that updating i cannot affect the radius of j, i.e. $j \in N_k(t+1)$.

Therefore, from the above cases we conclude that if the radius of an agent is less than r before updating agent i, then it will remain the same at the next time step, i.e. $N_k(t) \subseteq N_k(t+1)$.

Next we show that if the radius of an agent j is greater than r, i.e. $j \in N_s(t)$ for some $s > r$, then after updating i, it cannot reduce to something less than or equal to r at the next time step. Again, we consider three cases:

- $P_j(t) \notin \{a, b\}$: In this case updating i does not affect the radius of j and thus player j will remain in its own radius set at the next time step, i.e. $j \in N_s(t+1)$.
- $P_j(t) = \{a\}$: In this case and after updating i, the radius of j not only does not decrease but also it may increase, since there are fewer resources of type $\{a\}$.
- $P_j(t) = \{b\}$: In this case the graphical distance between j and i must be at least r' (recall that r' is the new radius of player i after her update) as otherwise, updating i from $P_i(t) = \{a\}$ to $P_i(t+1) = \{b\}$ will not increase player i's radius to r'. Therefore, updating i cannot reduce $r_j(t)$ to r or less.

Finally, we note that at least one player (player i) will leave the set $N_r(t)$ in favor of $N_{r'}(t+1)$ at the next time step. Therefore, $n_r(t+1)$ will be at least one less than $n_r(t)$. These together show the relations in (6.2). □

Although Lemma 6.1 already shows that the CSR game is an generalized ordinal potential game, to see this more explicitly and directly, let us define $\Phi(P(t)) = \sum_{i=1}^{D} n_i(t)n^{D-i}$. Now suppose that at time step t a player $i \in N_r(t)$ updates her radius from r to $r' > r$ by playing her strictly better response (thus $C_i(P(t)) > C_i(P(t+1))$). Using Lemma 6.1 we have

$$\begin{aligned}\Phi(P(t)) - \Phi(P(t+1)) &= \alpha n^{D-r} + \sum_{i=r+1}^{D} \big(n_i(t) - n_i(t+1)\big)n^{D-i}\\ &\geq n^{D-r} + \sum_{i=r+1}^{D} \big(n_i(t) - n_i(t+1)\big)n^{D-i}\\ &> n^{D-r} - n \times n^{D-r-1} = 0,\end{aligned}$$

where the first inequality is because $\alpha \geq 1$, and the last inequality is because $\sum_{i=r+1}^{D} n_i(t+1) \leq n$ (note that the sum of all the entries of the radius vector at some time t equals n), and $\sum_{i=r+1}^{D} n_i(t) > 0$. By Definition 6.4, this shows that $\Phi(P(t))$ serves as a generalized ordinal potential function for the CSR game. As a result CSR game always admits a NE.

Now, using Lemma 6.1 one can obtain a naive upper bound for the convergence time of the best response dynamics to a NE. One can in fact note that since $\sum_{r=1}^{D} n_r(t) = n$ for all t, the maximum number of updating steps before any other best response move is possible by any player (i.e., reaching a NE) can be upper bounded by the number of non-negative integer solutions of $z_1 + z_2 + \ldots + z_D = n$, which is equal to $\binom{n+D-1}{D-1} = \mathcal{O}(n^D)$, where D is the diameter of the graph $\mathcal{G}$. This shows that the best response dynamics converges reasonably fast to a NE over networks of small diameter. For example, as it has been shown in [5, Corollary 10.11], Erdos-Renyi random graphs with n vertices where each edge appears with probability p, have with high probability a diameter at most 2 whenever $p \geq \sqrt{\frac{c \ln n}{n}}$ for $c > 2$. In particular by taking $p = \frac{1}{2}$ almost all such random graphs over n vertices have diameter at most 2. In addition, most of the real world social networks benefit from having a low diameter, a small-world phenomenon which states that most nodes can be reached from every other node by a small number of hops [6]. Therefore, the convergence speed of the best response dynamics for the CSR game over small-world social networks is fast. However, as we are interested in the worst-case scenario, our aim is to devise some distributed algorithms which converge fast over general networks and not only over networks of small diameter, which will be our focus in the remainder of this chapter.

6.4 Least Best Response Dynamics

In this section we introduce a new updating rule for the players in order to arrive at a NE more efficiently. In fact, a closer look into Lemma 6.1 shows that the speed of convergence of the best response dynamics is highly dependent on the radius of the updating agent at each time instant. As we saw earlier, defining $\Phi(P(t)) = \sum_{i=1}^{D} n_i(t)n^{D-i}$, one can see that $\Phi(P(t))$ is a decreasing function over the iterations of the strictly best response dynamics. In particular, the lower the radius of updating, the steeper would be the decrease in the value of the function $\Phi(\cdot)$. Since such a function is always nonnegative, the iterations of the best response dynamics must terminate to a minimizer of this function which is a valid NE. Based on this observation, and in order to incorporate the role of updating radii into our algorithm, we introduce a slightly different version of the best response dynamics as follows:

Algorithm 1 Least Best Response

Given a CSR game, at each time $t = 1, 2, \ldots$, and from the set of all players who want to update, we select a player with the least radius and let her update be based on her strictly best response. Ties are broken arbitrarily.

Before we proceed with the analysis of Algorithm 2, we note that at each iteration, the algorithm must find a player with the least updating radius who wants to deviate. This can be done efficiently in a distributed manner as follows:

Assume each player i holds a binary flag $b_i \in \{0, 1\}$, and an integer valued variable x_i. Between every two consecutive steps of Algorithm 2 (say times t and $t + 1$), the players set all of their flags to 1, and their variables to the corresponding radii at time t, i.e., $x_i = r_i(t)$. After that, each player shares her variable with her immediate neighbors and updates it to the minimum of her own variable and those received from others. One can easily see that after at most D steps (where D is the diameter of the network), all the players' variables become equal to $\min_{i=1,\ldots,n} r_i(t)$. Now, each player can compare her variable with her own radius, and if the former is smaller than the latter, she sets her binary flag to 0, and leaves it unchanged, otherwise. Therefore, after at most D steps, all the players with binary flags equal to 1 are valid candidates to update at the next time step $(t + 1)$ of Algorithm 2. To avoid tie breaking issues on which candidate to update, one can run a similar protocol, this time by setting the variables to indices of the players.[3] This causes all the remaining flags with value 1 change to zero except the one which belongs to a candidate with the smallest index. This unique player will then update her resource based on her strictly best response at the next step $t + 1$ of the Algorithm 2. Since, the running time of the above protocol is at most $2D$, the overall running time of Algorithm 2 in a fully distributed manner is at most $2D$ times of that in the centralized version.

[3] Even if the players do not know their own indicies, they can uniformly sample a real number from [0, 1] and regard it as their own indices, in which case with probability 1 indices will be distinct.

We now turn to the analysis of Algorithm 1. First, we have the following lemma:

Lemma 6.2 *Let $|O|$ denote the number of resources in the CSR game. Then, in the least best response dynamics, we have $r(t) \leq |O| - 1, \forall t$, where $r(t)$ is the radius of the updating agent at time step t.*

Proof The proof is by contradiction. Suppose that there exists a time instant t such that $r(t) \geq |O|$. Based on the least best response dynamics, this can happen if all the agents with radii smaller than $r(t)$ are playing their best response. Now assume that an agent i with $P_i(t) = \{a\}$ has radius $r(t) \geq |O|$ and she wants to update her resource. We claim that all the players on the shortest path between i and the closest agent to her who has the same resource, i.e., $j = \sigma_i(P(t), P_i(t))$ must have different resources. Otherwise, if ℓ and ℓ' are two players on this shortest path who have the same resource, then the radius of ℓ, denoted by $r_\ell(t)$, will be smaller than $r(t)$. Since all the agents who have distances at most $r_\ell(t)$ to ℓ will have distances smaller than $r(t)$ to i, none of them can have $\{a\}$ as their resource. This means that player ℓ can improve her radius by updating her resource to $\{a\}$, which contradicts the fact that all the agents with radii smaller than $r(t)$ are playing their best responses. Therefore, none of the agents on the shortest path between i and j (except i and j) and the node in which i wants to update to its resource can have the same resource. Since the total number of resources is $|O|$, we must have $r(t) + 1 \leq |O|$. □

Remark 14 As a result of Lemmas 6.1 and 6.2, one can easily see that the least best response dynamics converge to a NE after no more than $n^{|O|-1}$ steps.

Before stating one of our main results in this section (Theorem 6.2), we state and prove another lemma which will be useful in the proof of Theorem 6.2.

Lemma 6.3 *In the least best response dynamics, suppose that a player $i \in N_r(t)$ updates her resource from $P_i(t) = \{a\}$ to $P_i(t+1) = \{b\}$ and thus increases her radius to r'. If there exists a player $\ell \in \cup_{j=1}^{r-1} N_j(t)$ who is playing her best response at time t but has an incentive to deviate at time $t+1$ (i.e., after updating player i), then we must have $|O| \geq 5$.*

Proof Let us assume that there exists such a player ℓ with $\ell \in N_k(t)$ for some $1 \leq k < r$. Also, let ℓ' be the nearest node to ℓ with the same resource, i.e. $\ell' = \sigma_\ell(P(t), P_\ell(t))$ and set $P_\ell(t) = P_{\ell'}(t) = \{c\}$ for some resource c. Note that in order for player i's deviation to affect player ℓ, she must be within a graphical distance of at most k from ℓ, i.e. $d_{\mathcal{G}}(i, \ell) \leq k$. Furthermore, we must have $d_{\mathcal{G}}(i, \ell) + k \geq r'$, otherwise, ℓ would be able to deviate to $\{b\}$ and increase her radius even before updating i (which is in contradiction with the updating rule in the least best response dynamics). Combining these relations, we get $k \geq \frac{r'}{2}$. Overall we have

$$1 \leq d_{\mathcal{G}}(i, \ell) \leq k < r < r', \quad k \geq \frac{r'}{2}. \tag{6.3}$$

Since k is an integer, we conclude that $k \geq 2$.

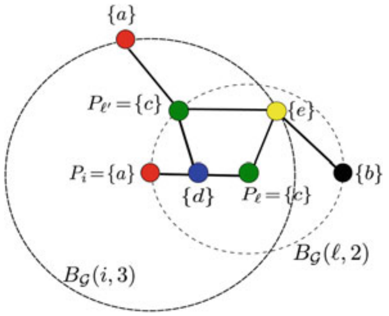


Fig. 6.3 Realization of Lemma 6.3 with minimum number of resources

Next we note that $c \notin \{a, b\}$. For if $c = b$, then there is no incentive for i to deviate from a to b. Also, if $c = a$, then $r \leq d_{\mathcal{G}}(i, \ell)$ which is in contradiction with (6.3). Moreover, since we know that agent ℓ was playing her best response before updating i, it means that all the resources have appeared at least once in $B_{\mathcal{G}}(\ell, k)$. In addition, since updating i from $\{a\}$ to $\{b\}$ creates this incentive for ℓ to deviate, the only way such a deviation can happen is that player ℓ evicts $\{c\}$ from her cache and inserts $\{a\}$. But this will lead to an increase in her radius from k to something larger, in which case it is not hard to see that the minimum number of resources for realizing such a situation is to have $d_{\mathcal{G}}(i, \ell) = k = 2, r = 3, r' = 4$ and with at least 5 different resources, as shown in Fig. 6.3. Thus $|O| \geq 5$ and this proves the Lemma. □

In the following theorem we now have the result that when the number of resources is at most 4, the least best response dynamics converges to a NE in linear time. This improves the convergence speed of an earlier algorithm in [1] for the case of two resources by a factor of n^2. Moreover, as we will see in Sect. 6.10, there is a close relationship between proper coloring of graphs and the NE points of the CSR game. While a proper coloring of a graph with only 2 colors can be done efficiently, with 3 or more colors it is an NP-hard problem. This may lead to the question of whether finding a NE in the CSR game when the number of resources is more than 2 can be done efficiently or not. As a result of the following theorem we will see that finding a NE in the CSR game with more than 2 resources can still be done efficiently.

Theorem 6.2 *In the CSR game with n players and $|O| \leq 4$, the least best response dynamics will converge to a NE in no longer than $3n$ steps.*

Proof Let us assume that at time t an agent $i \in N_r(t)$ who has radius r updates its resource from $P_i(t)$ to $P_i(t+1)$, and therefore moves to $N_{r'}(t+1)$ for some $r' > r$. Based on the least best response, i is one of unsatisfied agents at time t who has the least radius, which means that all the players in $\cup_{k=1}^{r-1} N_r(t)$ are playing their best response with respect to $P(t)$. On the other hand, since we have assumed $|O| \leq 4$, by Lemma 6.3 one can see that none of the players in $\cup_{k=1}^{r-1} N_r(t)$ has an incentive to deviate after player i's update. In other words, under the least best response dynamics if at some generic time t the minimum updating radius is r, then all the agents who have radii less than r will still play their best responses. Moreover, using Lemma 6.1, one can see that when the radius of updating at time t is r, after at most $n_r(t)$ steps the radius of updating will increase by at least 1 (because all the agents with radii less than r will stay with their best responses, and they will not move away from their own radius sets). Since by Lemma 6.2, r can be at most $\min\{|O|-1, D\} \leq 3$, and $n_r(t) \leq n$ for all t, we conclude that after at most $3n$ steps the dynamics will reach a NE. □

The following theorem now says that the least best response dynamics converges fast on relatively dense graphs.

Theorem 6.3 *In the CSR game with n agents and with $d_{\min} \geq |O|$, the least best response dynamics will converge to a NE in no longer than n^2 steps, where $d_{\min}$ denotes the minimum degree of the graph.*

Proof Consider an arbitrary player i. When $d_{\min} \geq |O|$, by the pigeon hole principle at least two of the neighbors of player i, say j and j', will have the same resources and hence their radii will be at most 2. We claim that the maximum updating radius in the sequence of least best response dynamics can be at most 2. Otherwise, if at some time t a player i with radius $r \geq 3$ wants to deviate by changing her resource from $P_i(t) = \{a\}$ to $P_i(t+1) = \{b\}$, this means that $B_{\mathcal{G}}(i, r)$ contains no resources of type $\{b\}$ at time step t. Since both j and j' are immediate neighbors of i, and $r \geq 3$, this implies that player j could strictly increase her radius by updating to $\{b\}$ at time t. But we know that the updating radius at time t is $r(t) \geq 3$ while j is an agent with a smaller radius of 2 who wants to deviate. This is in contradiction with the updating rule of Algorithm 2.

Finally, we argue that the maximum number of strictly best response updates with radii $r \leq 2$ is upper bounded by n^2. This is because by Lemma 6.1, any update of radius $r \leq 2$ will strictly decrease the first two coordinates of the radius vector given in Lemma 6.1 in a lexicographically order. Since the total number of such vectors restricted only on their first two coordinates are bounded above by n^2, namely, $(n, 0, *, \ldots, *)$, $(n-1, 1, *, \ldots, *)$, $\ldots$, $(0, 1, *, \ldots, *)$, $(0, 0, *, \ldots, *)$, this shows that after no more than n^2 steps the least best response dynamics will terminate to a profile where no other strictly better response is possible, i.e., a NE of the game. □

In fact, one can slightly generalize the result of Theorem 6.3 to the case where $\frac{|O|}{d_{min}} \leq c$ for some constant c, and obtain a polynomial time algorithm for finding a pure Nash equilibrium using least best response dynamics. We only mention this result in the following theorem:

Theorem 6.4 *The least best response dynamics reach an equilibrium in no longer than $n^{\frac{9|O|}{d_{min}}}$ steps, where d_{min} denote minimum degree of the network.*

According to Theorems 6.2 and 6.3, the least best response algorithm can provide polynomial time search for finding a NE in the CSR game when the number of resources is limited or when the underlying network is dense with respect to the number of resources; however, in general networks, the number of best responses to reach an equilibrium can be exponentially large $\mathcal{O}(n^D)$. To address this issue, in the next sections, we first study the price of anarchy (PoA) of the CSR game over general networks, and then leverage it to provide a constant approximation algorithm for optimal allocations and NE points of the CSR game which works in quasi-polynomial time over general networks, without any additional assumption on the structure of the network. Roughly speaking, this algorithm reduces the time complexity of being close to an equilibrium from the naive search of $\mathcal{O}(n^D)$ to $\mathcal{O}(n^{\ln D})$.

6.5 Price of Anarchy for the CSR Game

In this section, we show that all the NE points of the CSR game are almost as good as the global optimal allocation, i.e., an allocation profile which minimizes the total sum of the costs of the players.

Theorem 6.5 *The price of anarchy in the CSR game is bounded from above by 3.*

Proof For an arbitrary NE P^* and a specific node i with equilibrium radius r_i^*, i.e., $r_i^* = d_{\mathcal{G}}(i, \sigma_i(P^*, P_i^*))$, we note that all the resources must appear at least once in $B_{\mathcal{G}}(i, r_i^*)$. In fact, if a specific resource is missing in $B_{\mathcal{G}}(i, r_i^*)$, then node i can increase its radius by updating its current resource to that specific resource, thereby decreasing its cost (Remark 13). But this is in contradiction with P^* being an NE (Fig. 6.4). Now, given the equilibrium profile P^*, let us define $\hat{r}_i$ to be the smallest integer such that $B_{\mathcal{G}}(i, \hat{r}_i)$ contains at least two resources of the same type, i.e.,

$$\hat{r}_i = \min\Big\{r \in \mathbb{N} : \ \forall j, k \in B_{\mathcal{G}}(i, r-1), P_j^* \neq P_k^*, \text{and } \exists j_0, k_0 \in B_{\mathcal{G}}(i, r), P_{j_0}^* = P_{k_0}^*\Big\}.$$

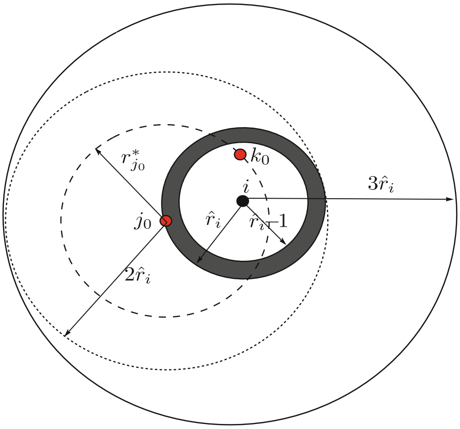


Fig. 6.4 Illustration of resource allocation in NE P^*. Note that $r_{j_0}^*$ denotes the radius of node j_0 at Nash equilibrium P^*, i.e., the graphical distance between node j_0 and the closest node to her which has the same resource. Thus $r_{j_0}^*$ is upper bounded by the distance between j_0 and k_0. The ball $B_{\mathcal{G}}(i, \hat{r}_i - 1)$ contains only resources of different types

Now we claim that all the resources must appear at least once in $B_{\mathcal{G}}(i, 3\hat{r}_i)$. To see this and by the above definition, let $j_0 \neq k_0 \in B_{\mathcal{G}}(i, \hat{r}_i)$ be such that $P^*_{j_0} = P^*_{k_0}$. This means that the equilibrium radius of node j_0, i.e., $r^*_{j_0}$ in P^* is at most $d_{\mathcal{G}}(j_0, i) + d_{\mathcal{G}}(i, k_0) \leq 2\hat{r}_i$. On the other hand, by the argument at the beginning of the proof, all the resources must appear at least once in $B_{\mathcal{G}}(j_0, 2\hat{r}_i)$. But since $B_{\mathcal{G}}(j_0, 2\hat{r}_i) \subseteq B_{\mathcal{G}}(i, 3\hat{r}_i)$, this shows that $B_{\mathcal{G}}(i, 3\hat{r}_i)$ must include all the resources at least once. Next, let us denote an optimal allocation profile by P^o, and the cost of node i in the optimal allocation and at NE by $C_i(P^o)$ and $C_i(P^*)$, respectively. Now for the Nash equilibrium P^*, and since by the definition of $\hat{r}_i$ there are no two similar resources in $B_{\mathcal{G}}(i, \hat{r}_i - 1)$, and all the resources appear at least once in $B_{\mathcal{G}}(i, 3\hat{r}_i)$, we can write

$$\begin{aligned} C_i(P^*) = \sum_{o \in O \setminus \{P_i\}} d_{\mathcal{G}}(i, \sigma_i(P^*, o)) \leq & \sum_{j \in B_{\mathcal{G}}(i, \hat{r}_i - 1)} d_{\mathcal{G}}(i, j) \\ & + 3\hat{r}_i \big(|O| - 1 - |B_{\mathcal{G}}(i, \hat{r}_i - 1)|\big). \end{aligned} \tag{6.4}$$

On the other hand, for the cost of node i in the optimal allocation P^o, we can write

$$\begin{aligned} C_i(P^o) = \sum_{o \in O \setminus \{P_i\}} d_{\mathcal{G}}(i, \sigma_i(P^o, o)) \geq & \sum_{j \in B_{\mathcal{G}}(i, \hat{r}_i - 1)} d_{\mathcal{G}}(i, j) \\ & + \hat{r}_i \big(|O| - 1 - |B_{\mathcal{G}}(i, \hat{r}_i - 1)|\big), \end{aligned} \tag{6.5}$$

where the inequality holds since node i has to pay at least $\sum_{j \in B_{\mathcal{G}}(i, \hat{r}_i - 1)} d_{\mathcal{G}}(i, j)$ for the first $|B_{\mathcal{G}}(i, \hat{r}_i - 1)|$ closest resources, and to pay at least $\hat{r}_i$ for the remaining $(|O| - 1 - |B_{\mathcal{G}}(i, \hat{r}_i - 1)|)$ resources. By comparing relations (6.4) and (6.5), it is not hard to see that $\frac{C_i(P^*)}{C_i(P^o)}$ is at most

$$\frac{\sum_{j \in B_{\mathcal{G}}(i, \hat{r}_i - 1)} d_{\mathcal{G}}(i, j) + 3\hat{r}_i \big(|O| - 1 - |B_{\mathcal{G}}(i, \hat{r}_i - 1)|\big)}{\sum_{j \in B_{\mathcal{G}}(i, \hat{r}_i - 1)} d_{\mathcal{G}}(i, j) + \hat{r}_i \big(|O| - 1 - |B_{\mathcal{G}}(i, \hat{r}_i - 1)|\big)},$$

which is bounded from above by 3. Since node i and the equilibrium P^* were chosen arbitrarily, for every equilibrium P^* and for all $i \in V(\mathcal{G})$, we have $C_i(P^*) \leq 3C_i(P^o)$. Summing both sides of this inequality over all $i \in V(\mathcal{G})$, we get $C(P^*) = \sum_i C_i(P^*) \leq 3\sum_i C_i(P^o) = 3C(P^o)$. Since we have this inequality for all possible Nash equilibria, and using the definition of the price of anarchy, we have $PoA = \frac{\max_{P^* \in NE} C(P^*)}{C(P^o)} \leq 3$. □

Next, in the following example, we provide an illustration of the fact that the price of anarchy of the CSR game can actually be arbitrarily close to 2 for some networks.

Example 6.1 Given arbitrary positive integers m, and with $|O| \geq 2$, let us consider a network of $n := (m + 1)(|O| - 1)$ nodes as shown in Fig. 6.5. The bottom part of the network is composed of a clique of $|O| - 1$ nodes, and the top part forms an independent set of $m(|O| - 1)$ nodes which are all connected to the bottom part. As it can be seen in Fig. 6.5, the left figure constitutes a pure NE for the CSR game with a

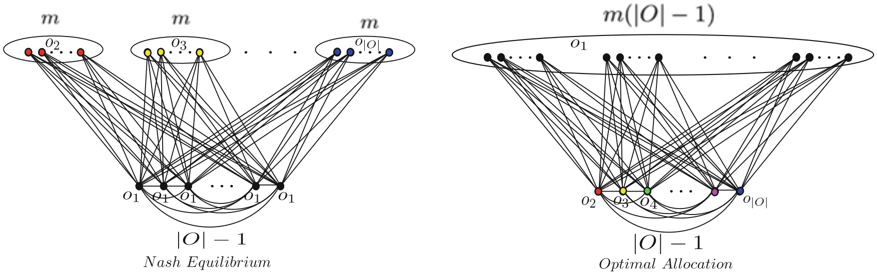


Fig. 6.5 Illustration of the resource allocation for an NE (*left figure*), and an optimal allocation (*right figure*) in Example 6.1. Note that in this example the optimal allocation is itself an NE

total cost of $C(P^*) = m(|O|-1)(2|O|-3) + (|O|-1)^2$. Further, it can easily be seen that the right figure illustrates an optimal allocation (which is also a NE), where the cost of each node is $|O|-1$, and hence the total optimal cost equals $C(P^o) = (m+1)(|O|-1)^2$. Thus, we have $\frac{C(P^*)}{C(P^o)} = 2 - \left(\frac{m+|O|-1}{(m+1)(|O|-1)}\right)$. This shows that for large m and $|O|$, the price of anarchy of the CSR game over such networks can be arbitrarily close to 2.

Here we note that there could be possibly other networks with price of anarchy even higher than or equal to 2, and the above example provides only one such network that we could think of [7].

6.6 Approximation Algorithm for the CSR Game

In this section we show that for the CSR game over general networks one can reach an allocation profile with total cost lying within a constant factor of that in an optimal allocation (and hence any NE) in only quasi-polynomial time. Toward that aim, we consider the following updating dynamics:

Algorithm 2 ϵ-Best Response

Given a network $\mathcal{G} = ([n], \mathcal{E})$, a real number $\epsilon > 1$, a set of available resources O, and an arbitrary initial allocation profile $P(0)$, at every time instance we select an agent i who can increase its radius $r_i(t)$ by a factor of at least ϵ, i.e., $r_i(t+1) \geq \epsilon r_i(t)$, and let her play her best response (ties are broken arbitrarily).

The following example illustrates how the ϵ-best response algorithm works.

Example 6.2 Let $\epsilon = 2$, and consider a network of 10 nodes and 4 available resources $O = \{o_1, o_2, o_3, o_4\}$. The initial profile of allocated resources has been illustrated in Fig. 6.6. Let us assume that at the first time instant node i has been selected by the

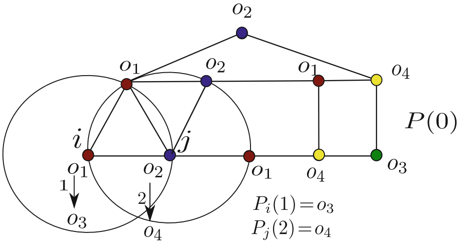


Fig. 6.6 Illustration of the resource allocation in the ϵ-best response algorithm (Example 6.2) for $\epsilon = 2$ and $|O| = 4$

algorithm. Since the radius of node i in the initial profile is 1, i.e., $r_i(0) = 1$, among the resources whose distances to node i are at least $\epsilon r_i(0) = 2 \times 1$ (in this example $\{o_3, o_4\}$), she will play her best response, i.e., o_3. Therefore, $P_i(1) = o_3$. Now at the second time instant, and given that node j is selected, since $r_j(1) = 1$, among the resources whose distances to j are at least $\epsilon r_j(1) = 2 \times 1$ (in this example only $\{o_4\}$), she will play her best response, i.e., $P_j(2) = o_4$. Note that if the set of available resources where an agent is supposed to choose her best resource from is empty at some time instant, then she will not update.

In the following lemma we show that indeed the ϵ-best response algorithm terminates in quasi-polynomial time.

Lemma 6.4 *Given a real number $\epsilon > 1$, the ϵ-best response algorithm terminates after at most $\mathcal{O}\left(n^2 D^{\log_\epsilon n}\right)$ steps.*

Proof Let us denote the radii of the players at time step t by $r_1(t), \ldots, r_n(t)$. Given that at time step t player i with $P_i(t) = o_1$ wants to change to $P_i(t+1) = o_2$, it means that at time step t, there is no resource of type o_2 in $B(i, \epsilon r_i)$ (Fig. 6.7). Now, let us define a potential function $R(t)$ as

$$R(t) := \sum_{k=1}^{n} \frac{1}{(r_k(t))^{\log_\epsilon n}} = \sum_{k=1}^{D} \frac{n_k(t)}{k^{\log_\epsilon n}}, \tag{6.6}$$

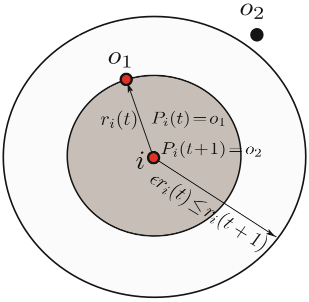


Fig. 6.7 Illustration of updating node i in Lemma 6.4. Note that there are no other resources of type o_2 in $B_{\mathcal{G}}(i, r_i(t+1))$

where the equality holds since by definition of $n_k(t)$, exactly $n_k(t)$ of the terms in $\sum_{k=1}^{n}\frac{1}{(r_k(t))^{\log_\epsilon n}}$ are equal to $\frac{1}{k^{\log_\epsilon n}}$. Moreover, one can easily see that $R(\cdot)$ is a nonnegative function which is upper bounded by n. We will show that after each time of running the dynamics, the value of the potential function given in (6.6) decreases by at least $\frac{1}{n\|\{r_{\max}(t)\}\|_\infty^{\log_\epsilon n}}$, where $\|\{r_{\max}(t)\}\|_\infty = \max_{t\geq 0}\max_{i\in[n]} r_i(t)$. To see this, let us assume that node i updates its radius from $r_i(t)$ to $r_i(t+1)\geq \epsilon r_i(t)$. Therefore, for some $k\in[n]$ we must have one of the following three cases:

- If $k=i$, then $r_i(t+1)\geq \epsilon r_i(t)$. This holds because of the dynamics rule. In other words, node i updates its radius from $r_i(t)$ to $r_i(t+1)$ if and only if $r_i(t+1)\geq \epsilon r_i(t)$.
- If $r_k(t)<r_i(t+1)$, then $r_k(t+1)\geq r_k(t)$. To see this, note that if $r_k(t)<r_i(t+1)$, then $P_k(t)\neq o_2$. Moreover, either $P_k(t)=o_1$, where in this case by updating $P_i(t)$ from $P_i(t)=o_1$ to $P_i(t+1)=o_2$, the radius of node k does not decrease, i.e., $r_k(t+1)\geq r_k(t)$, or $P_k(t)\neq o_1$, where in this case the radius of node k remains the same, i.e., $r_k(t+1)=r_k(t)$.
- If $r_k(t)\geq r_i(t+1)$, then $r_i(t+1)\leq r_k(t+1)$. To show this, first note that if $P_k\neq o_1, o_2$, then the radius of node k does not change after node i updates, i.e., $r_k(t+1)=r_k(t)\geq r_i(t+1)$. Otherwise, either $P_k(t)=o_1$, which in this case $r_k(t+1)\geq r_k(t)\geq r_i(t+1)$ due to the fact that after update we have fewer number of o_1-resources, or $P_k(t)=o_2$ which in this case the radius of player k cannot decrease to less than the graphical distance between k and i, thus again leading to $r_k(t+1)\geq r_k(t)\geq r_i(t+1)$.

Using the fact that $r_i(t+1)\geq \epsilon r_i(t)$, and considering the above three possibilities, we have

$$\begin{aligned}
R(t+1)-R(t) &= \sum_{k=1}^{n}\left(\frac{1}{(r_k(t+1))^{\log_\epsilon n}}-\frac{1}{(r_k(t))^{\log_\epsilon n}}\right)\\
&= \frac{1}{(r_i(t+1))^{\log_\epsilon n}}-\frac{1}{(r_i(t))^{\log_\epsilon n}}\\
&\quad +\sum_{\{k\neq i: r_k(t)<r_i(t+1)\}}\left(\frac{1}{(r_k(t+1))^{\log_\epsilon n}}-\frac{1}{(r_k(t))^{\log_\epsilon n}}\right)\\
&\quad +\sum_{\{k: r_k(t)\geq r_i(t+1)\}}\left(\frac{1}{(r_k(t+1))^{\log_\epsilon n}}-\frac{1}{(r_k(t))^{\log_\epsilon n}}\right)\\
&\leq (\frac{1}{n}-1)\left(\frac{1}{(r_i(t))^{\log_\epsilon n}}\right)+0+\frac{|\{k: r_k(t)\geq r_i(t+1)\}|}{n(r_i(t))^{\log_\epsilon n}}\\
&\leq (\frac{1}{n}-1)\left(\frac{1}{(r_i(t))^{\log_\epsilon n}}\right)+\frac{n-2}{n(r_i(t))^{\log_\epsilon n}}\\
&= \frac{-1}{n(r_i(t))^{\log_\epsilon n}}\leq \frac{-1}{n\|\{r_{\max}(t)\}\|_\infty^{\log_\epsilon n}},
\end{aligned}$$

where in the second to last inequality, and without any loss of generality, we can consider $|\{k : r_k(t) \geq r_i(t+1)\}| \leq n-2$. In fact, we argue that at most for D instances we can have $|\{k : r_k(t) \geq r_i(t+1)\}| \geq n-2$, which does not really change the quasi-polynomial order of the termination time. Note that player i does not belong to the set $\{k : r_k(t) \geq r_i(t+1)\}$, thus $|\{k : r_k(t) \geq r_i(t+1)\}| \leq n-1$. Moreover $|\{k : r_k(t) \geq r_i(t+1)\}|$ cannot be equal to $n-1$ for more than D steps, since by Lemma 6.1 every time that $|\{k : r_k(t) \geq r_i(t+1)\}| = n-1$, after updating node i at the next time step the minimum index of the positive entries in $n(t+1)$ will increase by at least 1, which cannot happen for more than D steps.

Finally, since $R(\cdot)$ is upper bounded by n, $R(\cdot)$ cannot decrease by more than $n^2\|\{r_{\max}(t)\}\|_\infty^{\log_\epsilon n}$ times, which shows that the dynamics must terminate after at most $\mathcal{O}\Big(n^2\|\{r_{\max}(t)\}\|_\infty^{\log_\epsilon n}\Big)$ steps. Moreover, since for all $t = 0, 1, \ldots$, and $i \in [n]$, $r_i(t)$ cannot exceed the diameter D of the network, $\|\{r_{\max}(t)\}\|_\infty \leq D$. This shows that the dynamics must terminate after at most $\mathcal{O}\Big(n^2 D^{\log_\epsilon n}\Big)$ steps. □

Next we introduce a useful definition.

Definition 6.6 Given an undirected network $\mathcal{G} = ([n], \mathcal{E})$, and an allocation profile P, the ***resource-radius*** of any node i is the smallest positive integer γ_i such that all the resources appear at least once in $B(i, \gamma_i)$ for that given profile P.

Now we are ready to state the main result of this section, which says that the allocation profile reached after executing the ϵ-best response dynamics is a constant approximation of the optimal allocation in the CSR game. In Sect. 6.7 we provide a dual-approximation algorithm which runs in only polynomial time over *almost* symmetric networks but with slightly worse constant factor than the ϵ-best response dynamics.

Theorem 6.6 *The ϵ-best response algorithm provides a $(2\epsilon+1)$-approximation of the optimal allocation after at most $\mathcal{O}\Big(n^2 D^{\log_\epsilon n}\Big)$ steps, i.e., $\frac{C_i(\overline{P})}{C_i(P^o)} \leq 2\epsilon+1, \forall i \in [n]$, where P^o and $\overline{P}$ denote the optimal allocation and the allocation obtain at the end of the algorithm, respectively.*

Proof First, we note that by Lemma 6.4 the ϵ-best response algorithm terminates after at most $\mathcal{O}\Big(n^2 D^{\log_\epsilon n}\Big)$ steps. Let us denote the allocation profile at the end of the algorithm by $\overline{P}$. Therefore, we only need to show that the final profile $\overline{P}$ is indeed within a constant factor of the optimal allocation. For any arbitrary $i \in [n]$, suppose that γ_i is the resource-radius of node i at the final profile $\overline{P}$. We now claim that $B_{\mathcal{G}}(i, \frac{\gamma_i-1}{2\epsilon+1})$ contains only different resources. To see this, let us assume that to the contrary there are two nodes $j, j' \in B_{\mathcal{G}}(i, \frac{\gamma_i-1}{2\epsilon+1})$ which share the same resource, i.e., $\overline{P}_j = \overline{P}_{j'}$. Since $j, j' \in B_{\mathcal{G}}(i, \frac{\gamma_i-1}{2\epsilon+1})$, $d_{\mathcal{G}}(j, j') \leq \frac{2(\gamma_i-1)}{2\epsilon+1}$. This shows that $r_j \leq \frac{2(\gamma_i-1)}{2\epsilon+1}$, and hence, $\epsilon r_j \leq \frac{2\epsilon(\gamma_i-1)}{2\epsilon+1}$. In particular, since $d_{\mathcal{G}}(j, i) \leq \frac{\gamma_i-1}{2\epsilon+1}$, $B_{\mathcal{G}}(j, \epsilon r_j) \subseteq B_{\mathcal{G}}(i, \frac{2\epsilon(\gamma_i-1)}{2\epsilon+1} + \frac{\gamma_i-1}{2\epsilon+1}) = B_{\mathcal{G}}(i, \gamma_i - 1)$. Since we assumed that $\overline{P}$ is the final allocation of the algorithm, every resource must appear at least once in $B_{\mathcal{G}}(j, \epsilon r_j)$, as otherwise node j can update its resource to another one which is outside of $B_{\mathcal{G}}(j, \epsilon r_j)$. But, by definition of resource-radius γ_i, we know that for the profile $\overline{P}$ there exists at least one resource which is missing in $B_{\mathcal{G}}(i, \gamma_i - 1)$. This is in contradiction with $B_{\mathcal{G}}(j, \epsilon r_j) \subseteq B_{\mathcal{G}}(i, \gamma_i - 1)$, which immediately implies that all the resources in $B_{\mathcal{G}}(i, \frac{\gamma_i - 1}{2\epsilon+1})$ must be different.

Finally, as we have shown above, for every player i, the final profile at the termination of Algorithm 1 has the property that all the vertices in $B_{\mathcal{G}}(i, \frac{\gamma_i-1}{2\epsilon+1})$ have different resources, while all the resources appear at least once in $B_{\mathcal{G}}(i, \gamma_i)$. Thus we can write

$$C_i(\overline{P}) \leq \sum_{j \in B_{\mathcal{G}}(i, \frac{\gamma_i-1}{2\epsilon+1})} d_{\mathcal{G}}(i, j) + \gamma_i \left(|O| - 1 - |B_{\mathcal{G}}(i, \frac{\gamma_i - 1}{2\epsilon + 1})| \right),$$

where the inequality holds due to the definition of γ_i. On the other hand, for the cost of node i in the optimal placement, we can write

$$C_i(P^o) \geq \sum_{j \in B_{\mathcal{G}}(i, \frac{\gamma_i-1}{2\epsilon+1})} d_{\mathcal{G}}(i, j) + (\frac{\gamma_i}{2\epsilon + 1}) \left(|O| - 1 - |B_{\mathcal{G}}(i, \frac{\gamma_i - 1}{2\epsilon + 1})| \right),$$

where the inequality holds since node i in the optimal allocation P^o has to pay at least $\sum_{j \in B_{\mathcal{G}}(i, \frac{\gamma_i-1}{2\epsilon+1})} d_{\mathcal{G}}(i, j)$ for the first $|B_{\mathcal{G}}(i, \frac{\gamma_i-1}{2\epsilon+1})|$ closest resources, and to pay an integer cost strictly larger than $\frac{\gamma_i-1}{2\epsilon+1}$ (at least $(\frac{\gamma_i}{2\epsilon+1})$), for the remaining $(|O| - 1 - |B_{\mathcal{G}}(i, \frac{\gamma_i-1}{2\epsilon+1})|)$ resources. By comparing the above two relations, it is not hard to see that for all $i \in [n]$, we have $C_i(\overline{P}) \leq (2\epsilon + 1)C_i(P^o)$. □

Corollary 6.1 *Replacing $\epsilon = e$, where $e := \exp(1)$ in the result of Theorem 6.6, one can easily see that Algorithm 2 gives us an allocation profile within a constant factor $1 + 2e$ of the optimal allocation after at most $\mathcal{O}\left(n^2 D^{\ln D}\right) = \mathcal{O}\left(n^{2+\ln D}\right)$ steps.*

Remark 15 The notion of approximation factor we are using in Theorem 6.6 is a multiplicative approximation factor, meaning that the cost of each player in the final outcome of the algorithm lies within a small multiplicative factor of that in a NE. This is slightly different from an ϵ-NE, where the cost of each player lies withing an additive ϵ error of that in a NE. Nevertheless, given that the costs are uniformly bounded above and the approximation factor can be made small enough, one can always recover an ϵ-NE from the former.

Remark 16 One can easily see that the allocation profile obtained at the end of Algorithm 1 is indeed an ϵ-approximation of an NE, meaning that no player can reduce its cost by a factor of more than $\frac{1}{\epsilon}$ (this can be seen using Remark 13 and the definition of Algorithm 1).

In fact the ϵ-best response algorithm can be implemented in an easy but somehow intuitive manner. Looking at the players of the CSR game as service providers, assume that each of them has to pay $\epsilon - 1$ fraction of his current saving as tax to the government if he wants to change his strategy by allocating a different resource (the current saving of each player can be interpreted as the radius of that player, because each player saves as much as his current radius due to the available resource in his cache). Now given that at time step t player i changes his resource from $P_i(t)$ to $P_i(t+1)$, his total income, i.e., $C_i(P(t)) - C_i(P(t+1)) - (\epsilon - 1)r_i(t)$ equals to

$$d_{\mathcal{G}}(i, \sigma_i(P(t), P_i(t+1))) - d_{\mathcal{G}}(i, \sigma_i(P(t), P_i(t))) - (\epsilon - 1)r_i(t) = d_{\mathcal{G}}(i, \sigma_i(P(t), P_i(t+1))) - \epsilon r_i(t).$$

Since player i will deviate only if his total income is positive, he will not deviate unless he can increase his radius to $d_{\mathcal{G}}(i, \sigma_i(P(t), P_i(t+1))) > \epsilon r_i(t)$. This is exactly what ϵ-best response algorithm expects from each player.

6.7 Dual Approximation Algorithm for the CSR Game

In this section, we establish results paralleling those in Sect. 6.6 using some duality in the CSR game. More precisely, in the ϵ-best response algorithm each player was trying to decrease his cost if his deviation brings a benefit of more than ϵ fraction of his current saving. We have already shown that such updates bring the entire system to some close neighborhood of the equilibrium points in quasi-polynomial time. In the following we establish a similar result which works even better (in polynomial time) when there is some uniformity or symmetry in the network.

In fact by Lemma 6.6, one would expect to see a "well-distributed" set of resources around each player. This observation suggests that in order to find an equilibrium one could devise an algorithm such that its evolution decreases the extra repetition of various resources around each player. Using this observation we introduce the following dynamics which aim to reduce the redundancy of similar resources (colors) around each player.

Algorithm 3 ϵ-Coloring

Given a network $\mathcal{G}$, a real number $\epsilon > 1$, a set of nonnegative integers $\{r_1, r_2, \dots, r_n\}$, and an arbitrary initial allocation profile $P(0)$, at every time instance we select an agent i whose ball $B_{\mathcal{G}}(i, r_i)$ contains at least two nodes with similar resources (colors), and we let any one of them (if possible) to update its resource to another one which did not appear in $B_{\mathcal{G}}(i, (2\epsilon + 1)r_i)$. Ties are broken arbitrarily.

Lemma 6.5 *The ϵ-coloring algorithm terminates after no more than $\mathcal{O}\Big(n^2|O|\left(\frac{r_{\max}}{r_{min}}\right)^{\log_\epsilon n}\Big)$ steps, where $r_{\max} = \max\{r_k : r_k > 0\}$ and $r_{\min} = \min\{r_k : r_k > 0\}$. Moreover, in the final profile and for any player i, either all the resources appear*

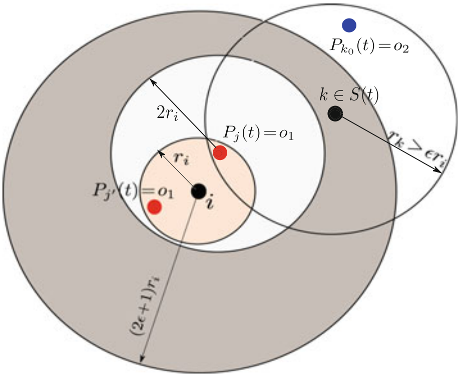


Fig. 6.8 Illustration of different nodes in the Lemma 6.5. Note that in this figure it is assumed that node j updates its resource at time $t+1$ from $P_j(t) = o_1$ to $P_j(t+1) = o_2$

at least once in $B_{\mathcal{G}}(i, (2\epsilon + 1)r_i)$, or every two players in $B_{\mathcal{G}}(i, r_i)$ have different resources.

Proof First, let us assume that the algorithm terminates with a profile $\overline{P}$. For any arbitrary but fixed node i, if all the resources are different in $B_{\mathcal{G}}(i, r_i)$, then the statement of the lemma holds. Otherwise, there are at least two nodes in $B_{\mathcal{G}}(i, r_i)$, namely j, j', that have the same color (Fig. 6.8). Since $d_{\mathcal{G}}(j, j') \le d_{\mathcal{G}}(j, i) + d_{\mathcal{G}}(i, j') \le 2r_i$, we have $B_{\mathcal{G}}(j, 2r_i) \subseteq B_{\mathcal{G}}(i, (2\epsilon + 1)r_i)$. Therefore, if at least one resource, namely o_2, is missing in $B_{\mathcal{G}}(i, (2\epsilon + 1)r_i)$, then node j can play a strictly better response by updating its current resource $\overline{P}_j$ to o_2. But, we assumed that $\overline{P}$ is the final profile of the algorithm, meaning that neither j nor j' has an incentive to deviate. This contradiction shows that if there are at least two nodes with similar resources in $B_{\mathcal{G}}(i, r_i)$, then all the resources must appear at least once in $B_{\mathcal{G}}(i, (2\epsilon + 1)r_i)$.

Next, we show that such dynamics terminate after no more than $\mathcal{O}\Big(n^2|O| \left(\frac{r_{\max}}{r_{min}}\right)^{\log_\epsilon n}\Big)$ updates. To see this, let $m_i(t)$ denote the number of different resources in $B_{\mathcal{G}}(i, r_i)$ at time step t, and define a potential function $M(t)$ to be

$$M(t) = \sum_{k:r_k>0} \frac{m_k(t)}{(r_k)^{\log_\epsilon n}}. \tag{6.7}$$

Note that $M(\cdot)$ is a nonnegative function which is upper bounded by $M(t) \le \sum_{k:r_k>0} \frac{|O|}{(r_k)^{\log_\epsilon n}} \le n\frac{|O|}{(r_{min})^{\log_\epsilon n}}$ for all t. We will show that after each time of running the dynamics, the value of the potential function given in (6.7) increases by at least $\frac{1}{n(r_{\max})^{\log_\epsilon n}}$. To see this, let us assume that two nodes $j, j' \in B_{\mathcal{G}}(i, r_i)$ have the same resources at time step t, i.e., $P_j(t) = P_{j'}(t) = \{o_1\}$, and at the next time step based on the dynamics, one of them, let us say node j, updates its resource from o_1 to o_2 which did not exist in $B_{\mathcal{G}}(i, (2\epsilon + 1)r_i)$ at time step t. Moreover, let $S(t) = \{k \in [n] : j \in B_{\mathcal{G}}(k, r_k),\ \exists k_0 \in B_{\mathcal{G}}(k, r_k),\ P_{k_0}(t) = o_2\}$. In other words, $S(t)$ is the set of all agents at time step t, which has agent j and at least one other agent with resource o_2 in their balls. We note that for any agent $k \in S(t)$, we must have $r_k > \epsilon r_i$. This is due to the fact that every agent with color o_2 is outside of $B_{\mathcal{G}}(i, (2\epsilon + 1)r_i)$. Therefore, since $d_{\mathcal{G}}(i, j) \leq r_i$, the graphical distance between j and every other agent with resource o_2 is at least $2\epsilon r_i$. Now since for every $k \in S(t)$, the ball $B_{\mathcal{G}}(k, r_k)$ contains both j and another resource of color o_2 whose distance to node j is at least $2\epsilon r_i$, r_k must be at least ϵr_i such that $B_{\mathcal{G}}(k, r_k)$ covers both j and one other resource of type o_2. Such a scenario has been illustrated in Fig. 6.8.

On the other hand, for any agent $k \in [n] \setminus S(t)$, and after updating node j from color o_1 to o_2 at time step $t + 1$, the value $m_k(t)$ is nondecreasing, i.e., $m_k(t) \leq m_k(t + 1), \forall k \in [n] \setminus S(t)$. This is because at time step t the ball $B_{\mathcal{G}}(k, r_k)$ either does not contain a resource of type o_2 or does not contain the node j. Putting together, at time $t + 1$, we have

- $m_i(t + 1) = m_i(t) + 1$,
- $m_k(t + 1) \geq m_k(t), \ \forall k \in [n] \setminus S(t)$,
- $m_k(t + 1) \geq m_k(t) - 1$ and $r_k \geq \epsilon r_i, \ \ \forall k \in S(t)$.

Now we can write

$$\begin{aligned}
M(t+1) - M(t) &= \sum_{k=1}^{n} \frac{m_k(t+1) - m_k(t)}{(r_k)^{\log_\epsilon n}} \\
&\geq \frac{1}{(r_i)^{\log_\epsilon n}} - \sum_{k \in S(t)} \frac{1}{(r_k)^{\log_\epsilon n}} \\
&\geq \frac{1}{(r_i)^{\log_\epsilon n}} - \sum_{k \in S(t)} \frac{1}{(\epsilon r_i)^{\log_\epsilon n}} \\
&\geq \frac{1}{(r_i)^{\log_\epsilon n}} \left(1 - \frac{n-1}{\epsilon^{\log_\epsilon n}}\right) \geq \frac{1}{n (r_{\max})^{\log_\epsilon n}},
\end{aligned}$$

where in the second last inequality we have used $|S(t)| \leq n - 1$, since at least $i \notin S(t)$. Therefore, $M(\cdot)$ cannot increase by more than $n^2|O| \left(\frac{r_{max}}{r_{min}}\right)^{\log_\epsilon n}$ times, which shows that the running time is at most $\mathcal{O}\left(n^2|O| \left(\frac{r_{max}}{r_{min}}\right)^{\log_\epsilon n}\right)$. □

Next we recall the definition of vertex density radius given in Definition 6.3. Note that by this definition τ_i is a fixed number that can be computed for each agent a priori and does not change as the allocation profile changes. Next, in the following theorem, we provide an algorithm for approximating any optimal allocation (and hence NE) of the CSR within a constant factor in only quasi-polynomial time.

Theorem 6.7 *In the CSR game with an initial allocation profile $P(0)$, let τ_k be the same as what has been defined in Definition 6.3, and consider the following two-stage process:*

- **Stage 1***: Run the ϵ-coloring algorithm for $r_k = \frac{\tau_k - 1}{2\epsilon + 1}$.*
- **Stage 2***: Initiating from the profile $\overline{P}$ obtained from Stage 1, run the ϵ-coloring algorithm for $r_k = \tau_k + 1$.*

Then, the allocation profile $\tilde{P}$ obtained at the end of Stage 2 is a constant approximation of optimal allocation and any NE after at most $\mathcal{O}\Big(n^2|O|\left(\frac{\tau_{max}}{\tau_{min}}\right)^{\log_\epsilon n}\Big)$ steps.

Proof Using the result of Lemma 6.5 for $r_k = \frac{\tau_k - 1}{2\epsilon+1}$, we get that at the end of the Stage 1 and for any arbitrary agent k, either all the resources appear at least once in $B_{\mathcal{G}}(k, \tau_k - 1)$, or every two nodes in $B_{\mathcal{G}}(k, \frac{\tau_k-1}{2\epsilon+1})$ have different resources. But by definition of τ_k, all the resources cannot appear in $B_{\mathcal{G}}(k, \tau_k - 1)$, which means that the second case must happen, i.e., at the termination of stage 1 and for any $k \in [n]$, all the resources in $B_{\mathcal{G}}(k, \frac{\tau_k-1}{2\epsilon+1})$ would be different. Now denoting the profile obtained at the end of Stage 1 by $\overline{P}$, we start the second stage by setting $P(0) = \overline{P}$, and $r_k = \tau_k + 1, \forall k \in [n]$. Similarly, using the result of Lemma 6.5, at the end of Stage 2 and for any arbitrary agent k, either all the resources appear at least once in $B_{\mathcal{G}}(k, (2\epsilon + 1)(\tau_k + 1))$, or every two nodes in $B_{\mathcal{G}}(k, \tau_k + 1)$ have different resources. But by the definition of τ_k, all the resources in $B_{\mathcal{G}}(k, \tau_k + 1)$ cannot be different. Therefore, for each k the first case must happen, i.e., at the termination of the algorithm and for any $k \in [n]$, all the resources must appear at least once in $B_{\mathcal{G}}(k, (2\epsilon + 1)(\tau_k + 1))$. Moreover, using Lemma 6.5 one can easily see that the total running time of both Stages 1 and 2 is at most $\mathcal{O}\Big(n^2|O|\left(\frac{\tau_{max}}{\tau_{min}}\right)^{\log_\epsilon n}\Big)$.

Next we argue that the property of having different colors in $\frac{\tau_k-1}{2\epsilon+1}$-neighborhood of any agent k which is inherited at the end of stage 1, will not be affected during the execution of the second stage. We show this by induction on the iteration steps $t = 0, 1, \ldots$ of the second stage. For the initial profile of the second stage $(\overline{P})$ which is set to be the final profile of the first stage, this property clearly holds. Now given that at the t-th iteration of the second stage, $B_{\mathcal{G}}(k, \frac{\tau_k-1}{2\epsilon+1}), \forall k \in [n]$ contains different resources, let us assume that at time $t + 1$ a node j which has the same resource as $j' \in B_{\mathcal{G}}(i, \tau_i + 1)$ for some i updates its resource from $P_j(t)$ to another resource o_2 which does not exist in $B_{\mathcal{G}}(i, (2\epsilon + 1)(\tau_i + 1))$. We claim that updating j from $P_j(t)$ to $P_j(t + 1) = o_2$ does not create any pair of nodes of the same resource in any of the balls $B_{\mathcal{G}}(k, \frac{\tau_k-1}{2\epsilon+1}), k \in [n]$. Otherwise, if it creates a pair of resources o_2 in some ball $B_{\mathcal{G}}(k_0, \frac{\tau_{k_0}-1}{2\epsilon+1})$, then that ball must contain both j and another resource o_2, which is at least $2\epsilon(\tau_i + 1)$ hops away from j. Therefore, $\frac{\tau_{k_0}-1}{2\epsilon+1} \geq \epsilon(\tau_i + 1)$, i.e., $\tau_i + 1 \leq \frac{\tau_{k_0}-1}{\epsilon(2\epsilon+1)}$. Thus

$$\begin{aligned}
d_{\mathcal{G}}(k_0, i) + (\tau_i + 1) &\leq d_{\mathcal{G}}(k_0, j) + d_{\mathcal{G}}(j, i) + (\tau_i + 1)\\
&\leq \frac{\tau_{k_0} - 1}{2\epsilon + 1} + 2(\tau_i + 1)\\
&\leq (\tau_{k_0} - 1)\left(\frac{1}{2\epsilon + 1} + \frac{2}{\epsilon(2\epsilon + 1)}\right) < \tau_{k_0} - 1.
\end{aligned}$$

This shows that $B_{\mathcal{G}}(i, \tau_i + 1) \subseteq B_{\mathcal{G}}(k_0, \tau_{k_0} - 1)$. Since by definition of τ_i we know that $B_{\mathcal{G}}(i, \tau_i + 1)$ must contain at least $|O|$ vertices, $B_{\mathcal{G}}(k_0, \tau_{k_0} - 1)$ also contains at least $|O|$ vertices. But this is in contradiction with the definition of τ_{k_0}.

Therefore, in step $t+1$ no pair of vertices with the same resource will be created in $B_{\mathcal{G}}(k, \frac{\tau_k - 1}{2\epsilon+1}), \forall k \in [n]$, which completes the induction process.

Since we have shown that for any agent k, the final profile at the termination of the algorithm, i.e., $\tilde{P}$, has this property that all the vertices in $B_{\mathcal{G}}(k, \frac{\tau_k - 1}{2\epsilon+1})$ have different resources, while all the resources appear at least once in $B_{\mathcal{G}}(k, (2\epsilon+1)(\tau_k+1))$. Thus we can write

$$C_k(\tilde{P}) = \sum_{o \in O \setminus \tilde{P}_k} d_{\mathcal{G}}(k, \sigma_k(\tilde{P}, o)) \leq \sum_{j \in B_{\mathcal{G}}(k, \frac{\tau_k - 1}{2\epsilon+1})} d_{\mathcal{G}}(k, j) \\ + (2\epsilon+1)(\tau_k+1)\left(|O| - 1 - |B_{\mathcal{G}}(k, \frac{\tau_k - 1}{2\epsilon+1})|\right).$$

On the other hand, for any allocation profile, and in particular for a NE P^* we always have

$$C_k(P^*) = \sum_{o \in O \setminus P_k^*} d_{\mathcal{G}}(k, \sigma_k(P^*, o)) \geq \sum_{j \in B_{\mathcal{G}}(k, \frac{\tau_k - 1}{2\epsilon+1})} d_{\mathcal{G}}(k, j) \\ + (\frac{\tau_k+1}{2\epsilon+1})\left(|O| - 1 - |B_{\mathcal{G}}(k, \frac{\tau_k - 1}{2\epsilon+1})|\right),$$

where the inequality holds since node k has to pay at least $\sum_{j \in B_{\mathcal{G}}(k, \frac{\tau_k - 1}{2\epsilon+1})} d_{\mathcal{G}}(k, j)$ for the first $|B_{\mathcal{G}}(k, \frac{\tau_k - 1}{2\epsilon+1})|$ closest resources, and to pay at least $\frac{\tau_k+1}{2\epsilon+1}$ for the remaining resources. Comparing the above two relations, one can easily see that for all $k \in [n]$, we have $C_k(\tilde{P}) \leq (2\epsilon+1)^2 C_k(P^*)$. □

Corollary 6.2 *The worst case running time of the allocation algorithm given in Theorem 6.7 is at most* $\mathcal{O}\left(|O| n^{2+\ln D}\right)$.

Proof Set $\epsilon = \exp(1)$. Since τ_{min} is the minimum over the set of all positive integers, $\tau_{min} \geq 1$. Moreover, we always have $\tau_{max} \leq D$. The result then follows by substituting these relations into the statement of Theorem 6.7. □

A closer look at the result of Theorem 6.7 shows that when the network is symmetric, then $\tau_{max} = \tau_{min}$, and hence, the running time of the algorithm shrinks to only $\mathcal{O}(n^2|O|)$. The same situation happens when the distribution of nodes in the network is somehow uniform, meaning that every node in the network observes almost the same number (up to a global constant factor) of nodes in different hops from it, i.e., $\frac{\tau_{max}}{\tau_{min}} = \mathcal{O}(1)$. As some examples of uniform networks one can list:

- **Grids (lattice)**: A graph with vertices in $\mathbb{Z}^k$ such that each node is adjacent to those whose Euclidean distance is 1 from it,
- **Hypercube**: A k dimensional hypercube is a graph with vertex set $\{0, 1\}^k$ such that two k-tuples are adjacent if and only if they differ in exactly one position,
- **Erdos-Renyi graphs**,

and many other uniform structures, in which case the running time of the above approximation algorithm will be polynomial.

6.8 ILP Formulation of Optimal Allocation and Nash Equilibrium

In this section we slightly deviate from our distributed algorithm setting in the previous section and focus on finding the optimal allocation and NE points of the CSR game in a centralized manner. For this purpose, we formulate the optimal allocation and NE points of the game as solutions of an integer linear program (ILP). This is important since relaxing such an optimization problem to a linear program (LP) allows us to approximate the optimal allocation and NE points in polynomial time, or to compute them using computer programs. Although such LP relaxations could be solved efficiently in polynomial time using the ellipsoid method or any other general tool for solving LPs (i.e., faster than the quasi-polynomial algorithm we provided in the previous section), still the integrality gap of these ILP's and their LP relaxations could be very large, resulting in uninteresting approximation factors which could possibly scale with the parameters of the problem (unlike the ϵ-best response algorithm which gives a good constant approximation). We leave the analysis of integrality gap for our ILP's formulations as an interesting future direction of research, and here we mainly focus on providing one such ILP formulation for the optimal allocation and NE points.

Toward this goal, let us introduce a vector $x_i \in \{0, 1\}^{|O|}$ for every player $i \in [n]$ such that its kth coordinate equals 1 if and only if player i allocates resource k. Therefore, each x_i is a binary vector which has exactly one entry equal to 1, and the rest are zeros. Moreover, for every $i \in [n]$, let $y_i \in \{0, 1\}^n$ be a vector such that its kth coordinate is 1 if and only if player i gets a resource from node k, i.e.,

$$y_{ik} = \begin{cases} 1 & \text{if player } i \text{ gets a resource from player } k \\ 0 & \text{else.} \end{cases}$$

(Note that since each player i gets one resource from himself, we have $y_{ii} = 1, \forall i \in [n]$). Finally for a given network $\mathcal{G}$, and every two players i and j, let $d_{ij} = d_{\mathcal{G}}(i, j)$ and $d_i = (d_{i1}, d_{i2}, \dots, d_{in})'$. Now we have the following theorem.

Theorem 6.8 *Every optimal allocation for the CSR game is a solution to the following integer linear program:*

$$\min \sum_{i=1}^{n} \sum_{k=1}^{n} d_{ik} y_{ik}$$

$$\sum_{j=1}^{|O|} x_{ij} = 1, \quad \sum_{i=1}^{n} z_{ijk} = 1,$$

$$z_{ijk} \leq x_{ij}, \ z_{ijk} \leq y_{ki}, \ x_{ij} + y_{ki} - z_{ijk} \leq 1,$$

$$y_{ik}, x_{ij}, z_{ijk} \in \{0, 1\}, y_{ii} = 1, \ \forall i, k \in [n], j \in [|O|].$$

Proof For every $i \in [n]$, we note that since every x_i has exactly one 1 in its coordinates, $\mathbf{1}'x_i = 1$. Stacking all the x_i, $i \in [n]$ as columns of a matrix X, we get $\mathbf{1}'X = \mathbf{1}'$. Furthermore, every y_i has exactly $|O|$ coordinates equal to 1 and the rest are zeros. Since $y_{ik} = 1$ means that player i gets a resource from player k and in total i needs exactly $|O|$ resources, the 1-entries in vectors x_k such that $y_{ik} = 1$ must lie in different coordinates, that is $\sum_{k=1}^{n} y_{ik}x_k = \mathbf{1}$, and hence, $Xy_i = \mathbf{1}$. Finally, it is not hard to see that if player i gets its needed resources from the players whose corresponding coordinates in y_i equal to 1, then the total cost for player i will be $C_i = d_i'y_i$, and hence, the total cost will be $\sum_{i=1}^{n} d_i'y_i$. Therefore, we can formulate the optimal allocation with the minimum total cost as the following integer program:

$$\begin{aligned}
&\min \sum_{i=1}^{n} d_i'y_i\\
&\mathbf{1}'X = \mathbf{1}', \quad Xy_i = \mathbf{1},\\
&y_i \in \{0,1\}^n, y_{ii} = 1, X \in \{0,1\}^{|O|\times n}.
\end{aligned} \tag{6.8}$$

In what remains, our goal is to only replace the nonlinear constraints with linear constraints. For every $i, k \in [n]$, and $j \in [|O|]$, let us define a new binary variable $z_{ijk} \in \{0,1\}$ to be $z_{ijk} = x_{ij}y_{ki}$. Since all of these variables are binary variables, it is not hard to see that $z_{ijk} = x_{ij}y_{ki}$ is equivalent to the following three linear inequalities:

$$z_{ijk} \le x_{ij}, \; z_{ijk} \le y_{ki}, \; x_{ij} + y_{ki} - z_{ijk} \le 1. \tag{6.9}$$

Substituting variables $x_{ij}y_{ki}$ with z_{ijk} in the integer program (6.8) and adding the inequalities given in (6.9) we obtain the desired result. □

Before we move to ILP formulation of the NE points of the CSR game, in the following theorem we use a simple probabilistic argument to upper bound the minimum social cost formulated in Theorem 6.8 in terms of the parameters of the problem. Surprisingly, this simple method can provide a very good bound on the total cost of the optimal allocation.

Theorem 6.9 *For a network $\mathcal{G}$ with vertex degrees d_i, $i \in [n]$, the optimal allocation cost for the CSR game is bounded above by* $n(2|O|-1) + |O|(|O|-1)\sum_{i=1}^{n}\left(1 - \frac{1}{|O|}\right)^{d_i}$.

Proof Let us assign with probability $\frac{1}{|O|}$ and independently a resource to any of the vertices of $\mathcal{G}$. Now for an arbitrary but fixed node i, $d_{\mathcal{G}}(i, \sigma_i(P, o))$ is a random variable denoting the graphical distance between player i and the player closest to him who has resource o under such a random resource allocation P. For any nonnegative integer r and any arbitrary but fixed resource $o \in O$, we have

$$\mathbb{P}\big\{d_{\mathcal{G}}(i,\sigma_i(P,o)) \geq r\big\} = \begin{cases} \left(1-\frac{1}{|O|}\right)^{|B_{\mathcal{G}}(i,r-1)|} & \text{if } 0 \leq r \leq D+1 \\ 0 & \text{else.} \end{cases}$$

Note that for $r = 0$ we have $B_{\mathcal{G}}(i, r-1) = \emptyset$, and hence, $|B_{\mathcal{G}}(i, r-1)| = 0$. Therefore, using the tail formula for expectation of discrete random variables (Lemma 2.11), the expected distance of player i from resource o is $\sum_{r=1}^{\infty}\mathbb{P}$ $(d_{\mathcal{G}}(i, \sigma_i(o, P)) \geq r)$. Therefore, the total expected cost of such a random assignment for all players and all resources is

$$\begin{aligned}\mathbb{E}[C(P)] &= \sum_{o\in O}\sum_{i=1}^{n}\sum_{r=0}^{\infty}\mathbb{P}(d_{\mathcal{G}}(i,\sigma_i(P,o)) \geq r)\\ &= |O|\sum_{i=1}^{n}\sum_{r=0}^{D+1}\left(1-\frac{1}{|O|}\right)^{|B_{\mathcal{G}}(i,r-1)|}\\ &= |O|\sum_{i=1}^{n}\left(1+\left(1-\frac{1}{|O|}\right)+\sum_{r=2}^{D+1}\left(1-\frac{1}{|O|}\right)^{|B_{\mathcal{G}}(i,r-1)|}\right)\\ &< n(2|O|-1)+|O|\sum_{i=1}^{n}\sum_{k=d_i+1}^{\infty}\left(1-\frac{1}{|O|}\right)^{k}\\ &= n(2|O|-1)+|O|(|O|-1)\sum_{i=1}^{n}\left(1-\frac{1}{|O|}\right)^{d_i}.\end{aligned}$$

This shows that there exists at least one resource assignment where the total cost is at most $n(2|O|-1)+|O|(|O|-1)\sum_{i=1}^{n}\left(1-\frac{1}{|O|}\right)^{d_i}$. □

Remark 17 One can easily check that for the line graph, the bound given in Theorem 6.9 is tight up to a constant factor [8].

Next we turn our attention to formulate NE points of the CSR game using an ILP. For this purpose, we benefit from the following lemma which characterizes Nash equilibria of the CSR game as maximizers of a function $f(\cdot)$.

Lemma 6.6 *Let r_i be the radius of agent i in an allocation profile P, and define $M_i(P)$ to be number of different resources in $B_{\mathcal{G}}(i, r_i)$. Then the function $f(\cdot) : \mathcal{P} \to \mathbb{R}$ defined by $f(P) = \sum_{i=1}^{n} M_i(P)$ achieves its maximum if and only if P is an NE of the CSR game.*

Proof First let us assume that P is an NE point of the CSR game. As we have seen earlier, this implies that for every player i all the resources must appear at least once in $B_{\mathcal{G}}(i, r_i)$, and hence $M_i(P) = |O|, \forall i \in [n]$. Therefore, $f(P) = n|O|$, which is the maximum possible value that could be taken by $f(\cdot)$ (note that in general we have $M_i(\cdot) \leq |O|, \forall i$). On the other hand, by Lemma 6.1 we know that the CSR game always admits a pure NE, and thus $\max_{P\in\mathcal{P}} f(P) = n|O|$. Therefore, if for some allocation profile we have $f(P) = n|O|$, this implies that $M_i(P) = |O|, \forall i \in [n]$, i.e., for every $i \in [n]$, all the resources appear at least once in $B_{\mathcal{G}}(i, r_i)$, which means that P must be an NE since no agent can increase its radius even further. □

Remark 18 Note that a major difference of the function $f(\cdot)$ from the generalized ordinal potential function $\Phi(\cdot)$ defined earlier is that unlike $\Phi(\cdot)$, the range of function $f(\cdot)$ is polynomially bounded (at most $|O|n$). This substantially alleviates characterizing the NE points using some variables and constraints whose total number is polynomially bounded in terms of the parameters of the CSR game.

Theorem 6.10 *Every NE of the CSR game over a network $\mathcal{G}$ of n nodes, diameter D, and pairwise distances $d_{ij}, i, j \in [n]$ can be obtained as a solution of the following ILP:*

$$
\begin{aligned}
&\max \sum_{i=1}^{n}\sum_{k=1}^{|O|} y_{ik}\\
&r_i \le D + (d_{ij} - D)\sum_{k=1}^{|O|} u_{ijk},\\
&w_{ijk} \le y_{ik} \le \sum_{j} w_{ijk},\ \ d_{ij}z_{ij} \le r_i < Dz_{ij} + d_{ij},\\
&u_{ijk} \le x_{ik},\ u_{ijk} \le x_{jk},\ x_{ik} + x_{jk} - u_{ijk} \le 1,\\
&w_{ijk} \le z_{ij},\ w_{ijk} \le x_{jk},\ z_{ij} + x_{jk} - w_{ijk} \le 1,\\
&r_i \in [D],\ x_{ij}, y_{ik}, z_{ik}, u_{ijk}, w_{ijk} \in \{0, 1\}, \forall i, j, k.
\end{aligned}
\tag{6.10}
$$

Proof In order to use the result of Lemma 6.6, we first formulate the radius of an agent i for a given allocation profile $P = (x_1, x_2, \dots, x_n)$. Let us choose an arbitrary player i and fix it. We claim that the radius r_i of agent i equals to

$$r_i = \min_j D + (d_{ij} - D)x_i'x_j. \tag{6.11}$$

In fact, it is not hard to see that $x_i'x_j = 1$ if and only if player j allocates the same resource as player i, and $x_i'x_j = 0$, otherwise. Therefore, $D + (d_{ij} - D)x_i'x_j$ equals to D whenever j does not have the same resource as i, and equals to d_{ij} whenever j has the same resource as i. Since $D \ge d_{ij}$, $\min_j D + (d_{ij} - D)x_i'x_j$ is equal to $\min_{j:x_j=x_i} d_{ij}$, which by definition is equal to the radius r_i of node i.

Next we count the number of different resources in r_i-neighborhood of node i. For this purpose, we define a new variable y_{ik} to be 1 whenever the kth resource o_k appears at least once in $B(i, r_i)$, and equals to 0, otherwise. Using this definition we can write $y_{ik} = \max_{j:d_{ij}\le r_i} x_{jk}$. Now let us define a variable $z_{ij} = \mathbb{I}_{\{d_{ij}\le r_i\}}$, using which we can write

$$y_{ik} = \max_{j:d_{ij}\le r_i} x_{jk} = \max_j \mathbb{I}_{\{d_{ij}\le r_i\}}x_{jk} = \max_j z_{ij}x_{jk}. \tag{6.12}$$

Moreover, we note that $z_{ij} = \mathbb{I}_{\{d_{ij} \le r_i\}}$ can be replaced by the following two equivalent linear inequalities

$$d_{ij} z_{ij} \le r_i < D z_{ij} + d_{ij}. \tag{6.13}$$

On the other hand, one can easily see that the number of different resources in $B_{\mathcal{G}}(i, r_i)$, i.e., $M_i(x_1, x_2, \ldots, x_n)$, is equal to $\sum_{k=1}^{|O|} y_{ik}$. Therefore, the function $f(x_1, \ldots, x_n)$ in Lemma 6.6 equals to $\sum_{i=1}^{n} \sum_{k=1}^{|O|} y_{ik}$. Hence by Lemma 6.6, maximizing this function subject to the constraints given in (6.11), (6.12) and (6.13), one can formulate the problem of obtaining NE points of the CSR game as follows:

$$\begin{aligned}
&\max \sum_{i=1}^{n} \sum_{k=1}^{|O|} y_{ik} \\
&r_i = \min_j D + (d_{ij} - D) x_i' x_j, \ \forall i, j \\
&y_{ik} = \max_j z_{ij} x_{jk} \\
&d_{ij} z_{ij} \le r_i < D z_{ij} + d_{ij} \\
&r_i \in [D], \ x_{ij}, y_{ij}, z_{ij} \in \{0, 1\}.
\end{aligned} \tag{6.14}$$

In the rest of the proof, our goal is to replace the non-smooth or nonlinear constraints in (6.14) using linear constraints.

- In order to remove the min-equality constraint $r_i = \min_j D + (d_{ij} - D) x_i' x_j$, we replace it by $r_i \le D + (d_{ij} - D) x_i' x_j, \ \forall j$, and call the new integer program IP$_{relaxed}$. This guarantees that $r_i \le \min_j D + (d_{ij} - D) x_i' x_j$. Hence, every feasible solution of (6.14) will be a feasible solution of IP$_{relaxed}$, which in turn shows that the optimal value in (6.14) is upper bounded by that in IP$_{relaxed}$. Next we argue that every optimal solution of (6.14) is indeed an optimal solution of IP$_{relaxed}$. Let us assume $x^* = \{x_{ij}^*\}$ is an optimal solution of IP$_{relaxed}$ with the corresponding variables $(y_{ij}^*, z_{ij}^*, r_i^*)$. If $r_i^* = \min_j D + (d_{ij} - D)(x_i^*)' x_j^*, \forall i$, then x^* is also a solution of (6.14) and we are done. But if for some i we have $r_i^* < \min_j D + (d_{ij} - D)(x_i^*)' x_j^*$, we can see that by increasing r_i^* to $\min_j D + (d_{ij} - D)(x_i^*)' x_j^*$, x^* remains a feasible solution for IP$_{relaxed}$. But r_i^* and y_{ij}^* are positively correlated, i.e., increasing r_i^* for some i does not decrease y_{ij}^* due to the constraints $d_{ij} z_{ij} \le r_i < D z_{ij} + d_{ij}$ and $y_{ik} = \max_j z_{ij} x_{jk}$. This shows that for fixed x^* by increasing r_i^* to $\min_j D + (d_{ij} - D)(x_i^*)' x_j^*$, the total objective value in IP$_{relaxed}$ will not decrease. Therefore, x^* and $\min_j D + (d_{ij} - D)(x_i^*)' x_j^*$ constitute a feasible solution for (6.14) where its objective value is at least as large as the optimal value in IP$_{relaxed}$, which we know is an upper bound for the optimal value of (6.14). This shows that x^* must be an optimal solution of (6.14) with the corresponding radii being $\min_j D + (d_{ij} - D)(x_i^*)' x_j^*$.
- We define new variables $u_{ijk} = x_{ik} x_{jk}$ and $w_{ijk} = z_{ij} x_{jk}$ by adding equivalent inequalities $u_{ijk} \le x_{ik}, \ u_{ijk} \le x_{jk}, \ x_{ik} + x_{jk} - u_{ijk} \le 1$, and $w_{ijk} \le z_{ij}, \ u_{ijk} \le$

x_{jk}, $z_{ij} + x_{jk} - w_{ijk} \leq 1$. This will remove the terms of the form of variable products.

- In order to replace max-equality constraints by linear constraints, we replace $y_{ik} = \max_j z_{ij}x_{jk} = \max_j w_{ijk}$ by $w_{ijk} \leq y_{ik}$, $\forall j$ and $y_{ik} \leq \sum_j w_{ijk}$. Note that $w_{ijk} \leq y_{ik}, \forall j$ guarantees that $\max_j w_{ijk} \leq y_{ik}$. Now since all the variables $y_{ik}, w_{ijk} \in \{0, 1\}$ are binary, $\max_j w_{ijk} \leq y_{ik}$ is always equivalent to $\max_j w_{ijk} = y_{ik}$, unless we have $w_{ijk} = 0, \forall j$. However, in this case the constraint $y_{ik} \leq \sum_j w_{ijk}$ becomes active and forces $y_{ik} = 0$.

Finally, incorporating all of the above changes into the IP (6.14), we obtain the ILP given in (6.10). □

Corollary 6.3 *Since by Lemma6.6 we know that the optimal solution in Theorem6.10 equals $n|O|$, we have $y_{ik} = 1, \forall i, k$. This reduces the optimization problem for finding a NE given in* (6.10) *to the following feasibility problem:*

$$
\begin{aligned}
& r_i \leq D + (d_{ij} - D)\sum_{k=1}^{|O|} u_{ijk}, \\
& 1 \leq \sum_j w_{ijk}, \quad d_{ij}z_{ij} \leq r_i < Dz_{ij} + d_{ij}, \\
& u_{ijk} \leq x_{ik}, \ u_{ijk} \leq x_{jk}, \ x_{ik} + x_{jk} - u_{ijk} \leq 1, \\
& w_{ijk} \leq z_{ij}, \ w_{ijk} \leq x_{jk}, \ z_{ij} + x_{jk} - w_{ijk} \leq 1, \\
& r_i \in [D], \ x_{ij}, z_{ij}, u_{ijk}, w_{ijk} \in \{0, 1\}, \ \forall i, j, k.
\end{aligned}
$$

6.9 Local Capacitated Selfish Replication (LCSR) Game

By a closer look at the structure of the CSR game, one can see that one of the drawbacks of such a model is the lack of local interactions among the players. In other words, based on the structure of the network and the number of resources, a player may have to travel a long distance in order to access some specific resource. This requires the players to have a global knowledge of the structure of the entire network, which can be impractical in real situations. This motivates us to consider in this section a slightly different version of the CSR game where the actions of the players are restricted to only their local neighborhoods. We refer to such a model as *local capacitated selfish replication* game, or simply *LCSR* game.

To start with, let ℓ be a constant which determines the depth of search for each player. We will consider a local version of the CSR game where for each player $i \in [n]$, the search area of the ith player for finding new resources is restricted to those players who are at most ℓ-hops away from him. Therefore, given an allocation profile P, the best action of each player can be totally determined by resources of his neighbors who are at most ℓ-hops from him. As an example for $\ell = 1$, each player takes into his account the resources of only his immediate neighbors. Now very similar to (6.1), we define the cost of player i to be

$$C_i^{(\ell)}(P) = \sum_{o \,\in \bigcup\limits_{j\in B_{\mathcal{G}}(i,\ell)} P_j} d_{\mathcal{G}}\big(i, \sigma_i(P,o)\big) + \theta_\ell\left(|O| - \Big|\bigcup_{j\in B_{\mathcal{G}}(i,\ell)} P_j\Big|\right), \tag{6.15}$$

where $\sigma_i(P,o)$ is $i's$ nearest node holding object o in allocation profile P, and θ_ℓ is a constant greater than ℓ which denotes the cost of having access to any resource farther than ℓ-hops from player i. From the above definition, it should be clear that LCSR game coincides with the original CSR game when $\ell = D$.

Theorem 6.11 *Any sequence of strict best response dynamics will lead to a pure NE of the LCSR game in only polynomial time $O(n^{\ell})$, where ℓ is the search depth constant of the game.*

Proof For an arbitrary but fixed player i, one can see that due to the cost function given in (6.15), all the resources which do not appear in $B_{\mathcal{G}}(i,\ell)$ incur the same cost of θ_ℓ for the player i. Therefore, there is no preference for player i to choose one over the other. As a result, the best response of any player in the LCSR game of depth search ℓ is totally characterized by the ℓth-hop neighborhood of that player. Note that as before we assume that each player has access to the set of all the resources but can only hold one resource in his cache. Therefore, the total set of resources accessible to a node does not change with the stage in the iteration and is always equal to O.

Next we note that any strict best response of the LCSR game is a strictly better response of the original CSR game. This is because if a player in the LCSR game can decrease his cost by playing his best response, using (6.15) and Remark 13 it means that he can strictly increase his radius. Therefore, by Lemma 6.1 every strict best response in the LCSR game will decrease the radius vector $n(t)$, lexicographically. Finally, employing the discussion at the beginning of the proof we can see that any strict best response update in the LCSR game must increase the radius of an agent whose radius before update was less than ℓ. This is because, if the radius of an agent in the LCSR game is more than ℓ, then by (6.15) that agent is already playing his best response and has no incentive to deviate. Therefore, every strict best response in the LCSR game must affect the first ℓ coordinates of the radius vector $n(t)$ and will decrease the entire vector $n(t)$, lexicographically. Since the total number of such events is at most n^{ℓ}, the LCSR game will reach a pure-strategy NE in no longer than n^{ℓ} steps. □

It would be interesting to consider the LCSR game when $\ell = 1$, i.e., when each player minimizes its cost with respect to only its immediate neighbors. In this case, the cost function of player i can be written as

$$C_i^1(P) = \left(\Big| \bigcup_{j \in \mathcal{N}(i)} P_j \Big| - 1 \right) + \theta_1 \left(|O| - \Big| \bigcup_{j \in \mathcal{N}(i)} P_j \Big| \right)$$
$$= \big(\theta_1 |O| - 1\big) - (\theta_1 - 1) \Big| \bigcup_{j \in \mathcal{N}(i)} P_j \Big|. \tag{6.16}$$

Since $\theta_1 > 1$, to minimize $C_i^1(P)$, player i must maximize $\big| \bigcup_{j \in \mathcal{N}(i)} P_j \big|$. In other words, he must choose a resource which is different from resources of his neighbors.

Now let us call an edge of $\mathcal{G}$ *monochromatic* if both of its endpoint players have the same resources, and let $E_i^{(1)}(P)$ denote the number of monochromatic edges emanating from node i, and $E^{(1)}(P) = \frac{1}{2} \sum_{i=1}^{n} E_i^{(1)}(P)$ be the total number of monochromatic edges in the network. One can see that $E^{(1)}(P)$ serves as a generalized ordinal potential function for the LCSR game with $\ell = 1$, meaning that after every strict best response of the player the value of $E^{(1)}(P)$ decreases by at least 1. Since in general $E^{(1)}(P)$ is bounded above by the total number of edges in the network, i.e., $O(n^2)$, thus the best response dynamics for LCSR ends with a NE after at most $O(n^2)$ steps. Nevertheless, using the result of Theorem 6.11 one can see that the best response dynamics lead to an NE in at most $O(n)$ steps. Although minimizing the potential function $E^{(1)}(P)$ does not yield a faster rate of convergence to an NE, the NE points obtained by minimizing $E^{(1)}(P)$ have an advantage that resources are better distributed around the players. For example let us consider a star network of n nodes and two resources o_1 and o_2, such that $n - 2$ of the leaves have resource o_1 and the last leaf has resource o_2 (note that we have assumed $\ell = 1$). Then in this case minimizing $E^{(1)}(P)$ forces the center node to choose resource o_2, while the natural best response allows the central player to choose either o_1 or o_2.

In fact, this duality between minimizing the monochromatic edges and minimizing the costs by playing a better response (one can think as maximizing the utilities) is not a coincidence. By extending the definition of monochromatic edges to monochromatic pairs one can define $E_i^{(r)}(P) = |\{(i, j) : d_{\mathcal{G}}(i, j) = r,\ P_j = P_i\}|$ to be the number of monochromatic pairs of length r emanating from node i, and $E^{(r)}(P) = \frac{1}{2} \sum_{i \in V} E_i^{(r)}(P)$ to be the total number of such monochromatic pairs of distance r from each other. Then one can see by a similar argument as in the proof of Lemma 6.1 ([9]) that after every strictly better response of any player the sorted vector $E(t) = (E^{(r)}(P(t)) : r \in [D])$ decreases lexicographically.

Finally, as we close this section, we want stress the tradeoff between localization in the LCSR game and the multiplicity of the equilibrium points. In fact, although reducing the depth of search in the LCSR game will reduce the complexity of the search for an equilibrium, it will increase multiplicity of the equilibrium points as well as the price of anarchy. As an example, for the line graph of n nodes, $|O| \geq 3$, and depth of search $\ell = 1$, one can see that every allocation profile without any chromatic edge is an NE.

6.10 CSR Game Versus Graph Coloring

In this section we discuss some of the similarities between coloring of graphs and equilibrium seeking in CSR games. Toward that end, let us assume that there is a polynomial time algorithm which can deliver a NE of the CSR game, which we denote by $P^* = (P_1^*, P_2^*, \dots P_n^*)$. Let us define $r_i^* = d_{\mathcal{G}}(i, \sigma_i(P^*, P_i^*)) - 1$ and $r^* = \min_{i=1,\dots,n} r_i^*$. Next, we define a new graph $\mathcal{G}^{(r^*)}$ to have the same set of vertices as $\mathcal{G}$, with an edge between two nodes in $\mathcal{G}^{(r^*)}$ if and only if the graphical distance between them in the original graph $\mathcal{G}$ is at most r^* (by convention we let $\mathcal{G}^{(0)}$ be the empty graph with no edges). If we consider the set of resources as the colors with $|O| = k$, based on our construction it is not hard to see that P^* is a k-proper coloring for $\mathcal{G}^{(r^*)}$ (a coloring of vertices of a graph with k colors is called proper if the end points of each edge in that coloring have different colors). Since $\mathcal{G} = \mathcal{G}^{(1)}$ and $\mathcal{E}(\mathcal{G}^{(0)}) \subseteq \mathcal{E}(\mathcal{G}^{(1)}) \subseteq \dots \subseteq \mathcal{E}(\mathcal{G}^{(r^*)})$, P^* is a k-proper coloring for $\mathcal{G}$ when $r^* > 0$. We have:

Proposition 6.12 *Let h be the largest integer such that $\mathcal{G}^{(h)}$ has a k-proper coloring, then there exists an equilibrium P^* for the CSR game such that $h = r^*$. Moreover, if the k-proper coloring in $\mathcal{G}^{(h)}$ is unique (up to relabeling of the colors), then it is an equilibrium for the CSR game.*

Proof Suppose that there is a k-proper coloring for $\mathcal{G}^{(h)}$. We use this allocation as an initial profile of the CSR game and start to update the agents' resources based on their best response dynamics. Since we know that the game is potential and by Lemma 6.1 the radius of the smallest ball will not decrease during this process, an NE can be reached such that the smallest radius of the nodes is at least as large as h. Furthermore, the smallest radius of nodes at the NE must be less than $h + 1$, as otherwise, the new equilibrium would be a k-proper coloring for $\mathcal{G}^{(h+1)}$, which is a contradiction. Therefore we must have $h = r^*$. Also, note that if Q is the unique k-coloring for $\mathcal{G}^{(h)}$, setting the initial profile to be Q in the least best response dynamics will lead to an equilibrium for the CSR game. Based on the above discussion this NE is a k-proper coloring for $\mathcal{G}^{(h)}$ which by assumption we know is unique. Thus, Q is an NE. □

Let us denote the set of NE points of the CSR game on the network $\mathcal{G}$ by $E(\mathcal{G})$ and the set of k-proper colorings of the graph $\mathcal{G}$ by $C(\mathcal{G})$. Using the same line of argument as in the proof of Proposition 6.12 we can see that if $C(\mathcal{G}) \neq \emptyset$, then starting from a coloring in $C(\mathcal{G})$ and applying the best response dynamics, it can be extended to an equilibrium which is still a valid k-proper coloring of $\mathcal{G}$. Therefore, if $C(\mathcal{G}) \neq \emptyset$, we must have $C(\mathcal{G}) \cap E(\mathcal{G}) \neq \emptyset$, and hence, we have the following corollary.

Corollary 6.4 *If $\mathcal{G}$ has a unique k-proper coloring up to relabeling of the colors, it must be a NE for the CSR game.*

As an example, let us consider two triangles which are sharing an edge and 3 different colors (resources) $\{a, b, c\}$. One possible 3-proper coloring for this network is to assign colors b and c to the incident nodes of the shared edge, and color a to the other two nodes. One can easily see that this coloring is unique up to relabeling of the colors, which by Corollary 6.4 must be NE of the CSR game. However, on the same network, if we assign color a to both end points of the shared edge, and b and c to the other two nodes we can obtain another NE which is not a proper coloring.

6.11 CSR Game with Different Capacities

Heretofore we had assumed that the capacity of each agent is exactly 1. In this section, we generalize our results to the case when this capacity is allowed to be more than 1, and possibly different for different agents. To be more precise, let us assume that agent i's capacity is $L_i \in \mathbb{N}$ where $1 \leq L_i \leq |O|$. Similarly, for a particular allocation profile $P = (P_1, P_2, \dots, P_n)$, with $|P_i| = L_i, i \in [n]$, we define the sum cost function of the ith player $C_i(P)$ as before by:

$$C_i(P) = \sum_{o \in O \setminus P_i} d_{\mathcal{G}}(i, \sigma_i(P, o)), \tag{6.17}$$

where $\sigma_i(P, o)$ is $i's$ nearest node holding o in P.

We next show that as before the resulting game is a generalized ordinal potential game and that one can find a NE of the game in polynomial time when the number of resources is limited.

Theorem 6.13 *The CSR game with different capacities admits a NE.*

Proof Starting from any allocation profile, and if necessary, by flipping the resources of each player in an arbitrary manner, we first make sure that each cache contains only different resources. Denote all the resources in the cache of player i at time t by $P_i(t) = \{o_{i_1}, o_{i_2}, \dots, o_{i_{L_i}}\}$, and assume that this player wants to update its cache to $P_i(t+1) = \{o'_{i_1}, o'_{i_2}, \dots, o'_{i_{L_i}}\}$ based on its strictly best response. Note that $P_i(t+1)$ can contain only different resources. Otherwise, replacing any one of the resources which is repeated multiple times in $P_i(t+1)$ by another one which has not appeared in that set, player i can strictly decrease its cost, contradicting the fact that $P_i(t+1)$ is the best response for player i.

Next, let us replace each player i with capacity L_i with L_i new sub-nodes as a clique, namely $i_1, i_2, \dots, i_{L_i}$, each with capacity 1. We treat all of these sub-nodes in the clique as node i and very naturally connect them to all the other sub-nodes where i was connected before. Figure 6.9 depicts such a scenario. We refer to this new game where each sub-node has capacity 1 as the unit cache size game. Using the definition of cost function given in (6.17), and since $P_i(t+1)$ was the strictly best response of player i in the original game, we must have

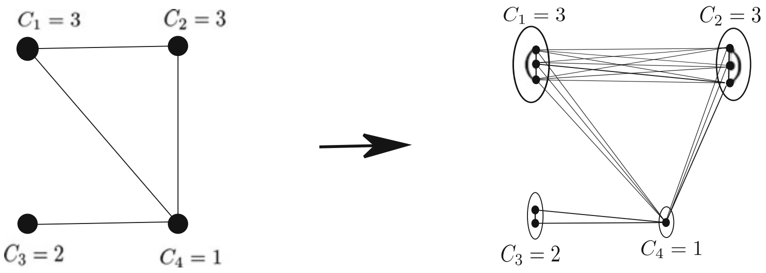


Fig. 6.9 An example of the original CSR game and its equivalent CSR game with unit size cache

$$\sum_{j=1}^{L_i} d_{\mathcal{G}}(i, \sigma_i(P(t), o'_{i_j})) > \sum_{j=1}^{L_i} d_{\mathcal{G}}(i, \sigma_i(P(t), o_{i_j})).$$

On the other hand, by symmetry of the sub-nodes associated with player i in the unit cache size game, we can always permute their resources among them without actually changing the unit cache size game. Therefore, without any loss of generality we can assume that the first k sub-nodes $i_1, \ldots, i_k$ have similar resources at times t and $t+1$, where $k = |P_i(t) \cap P_i(t+1)|$. Canceling the first k terms from both sides of the above inequality, we obtain

$$\sum_{j=k+1}^{L_i} d_{\mathcal{G}}(i, \sigma_i(P(t), o'_{i_j})) > \sum_{j=k+1}^{L_i} d_{\mathcal{G}}(i, \sigma_i(P(t), o_{i_j})),$$

where now $\{o_{i_{k+1}}, \ldots, o_{i_{L_i}}\} \cap \{o'_{i_{k+1}}, \ldots, o'_{i_{L_i}}\} = \emptyset$, due to the fact that these sets are representing $P_i(t) \setminus P_i(t+1)$ and $P_i(t+1) \setminus P_i(t)$, respectively. This shows that there exists at least one index j^* such that

$$d_{\mathcal{G}}(i, \sigma_i(P(t), o'_{i_{j^*}})) > d_{\mathcal{G}}(i, \sigma_i(P(t), o_{i_{j^*}})). \tag{6.18}$$

In particular, since $o'_{i_{j^*}} \in P_i(t+1) \setminus P_i(t)$, we must have $o'_{i_{j^*}} \notin P_i(t)$. Now since $o_{i_{j^*}}$ and $o'_{i_{j^*}}$ are the allocated resources to sub-node i_{j^*} at time steps t and $t+1$, respectively, this implies that in the unit cache size game the sub-node i_{j^*} can strictly decrease its cost by changing its resource from $o_{i_{j^*}}$ to $o'_{i_{j^*}}$. This shows that any strictly best response in the original game implies a strictly better response in its counter part unit cache size game. Since we already know that the unit cache size game is a generalized ordinal potential game (Lemma 6.1), it can admit at most finitely many strictly better responses. This shows that the original game with different capacities can also admit at most finitely many strictly best responses, which establishes the existence of a NE in such a game. □

Next in the following theorem we show that one can recover a NE for a CSR game with different capacities from its counterpart unit cache size game.

Theorem 6.14 *There exists a way to construct a NE for the CSR game from its counterpart unit cache size game. In particular, if $|O| \le L_{\min}d_{\min}$, where $L_{\min} = \min_{i=1\ldots n} L_i$, then there exists a polynomial time algorithm to find a NE of the CSR game with different capacities.*

Proof Let us consider a NE for the unit cache size game associated with the original CSR game with different capacities (by Lemma 6.1 such a NE exists). If in this NE all the sub-nodes which are representing some player i have different resources, then by stacking resources of the i-th clique as the cache content of player i, one can obtain an allocation profile which by a similar argument as in the proof of Theorem 6.13 must be a NE for the original game. Therefore, our goal in the remainder of the proof is to construct a NE for the unit cache size game without any multiplicity of resources within each clique.

For this purpose, we start from an initial profile where no two sub-nodes within a clique have similar resources. By applying the strictly best response dynamics on the unit cache size game, we claim that in none of the intermediate steps of the dynamics two resources of the same type will be created within a clique. Otherwise, let t be the first time such that a sub-node i_j updates its resource and creates multiplicity of resources within clique i. This update makes the radius of sub-node i_j equal to 1 (note that all the sub-nodes are connected in the i-th clique). But since that update was a strictly best response move for the sub-node i_j, this means that its radius before this move was less than 1, violating the fact that the radius is always a positive integer. Thus no strictly best response move will create a multiplicity of resources within a clique. Since we already know that the sequence of strictly best response dynamics terminates to a NE for the unit cache size game, we thus obtain a NE with no multiplicity of resources within each clique, as desired.

Finally, we note that in the unit cache size game, instead of n players we will now have $L = \sum_{i=1}^{n} L_i$ players. Also, the minimum degree in the new graph will be at least $d_{\min}L_{\min}$. Thus substituting L for n and $d_{\min}L_{\min}$ for $d_{\min}$ in Theorems 6.2 and 6.3 will give us a polynomial time algorithm L^2 to construct a NE in the unit cache size game with no multiplicity of resources within each clique, provided that $|O| \le L_{\min}d_{\min}$, and hence, a NE in the original game. □

By a closer look at the proof of Theorem 6.14, one can see that in order to find a NE efficiently for the CSR game with different capacities we would only require that the least best response dynamics converges efficiently on the equivalent CSR game with unit cache size. On the other hand, by Theorem 6.3 a sufficient condition for polynomial time convergence of the least best response dynamics in the CSR game with unit cache size is to have $d_{\min} \ge |O|$. This suggests that if we could increase the capacity of the nodes properly (which in turn will increase the minimum degree of nodes in the equivalent CSR game with unit cache size), then we would have a polynomial time algorithm for finding a NE in the game with higher capacities.

Based on the above observation, as a network designer we can ask the following question: given a network $\mathcal{G}$, what are the best nodes one can select so that by increasing their capacities one can guarantee that the least best response dynamics converges in polynomial time to a NE of the CSR game with higher capacities? In other words, it would be very appealing if increasing cache sizes of a few nodes in the network can guarantee an efficient algorithm for finding a NE. Therefore, in the following we formulate this problem as an integer programming problem. Relaxing this integer program to a linear program we find an approximate solution for the minimum extra capacities and their corresponding locations in the network for fast convergence to a NE in the CSR game with higher capacities.

Let us denote the capacity of node i by a variable x_i where $x_i \in \mathbb{N}$, and let $x := (x_1, \dots, x_n)'$. As noted in the proof of Theorem 6.14, by expanding the network and replacing node i with x_i identical nodes each of unit capacity, the degree of each of these new nodes would be $x_i - 1 + \sum_{j \in \mathcal{N}(i)} x_j$, where $\mathcal{N}(i)$ is the set of neighbors of node i in $\mathcal{G}$. Moreover, by Theorem 6.3 a sufficient condition for fast convergence of the least best response dynamics to a NE is

$$x_i - 1 + \sum_{j \in \mathcal{N}(i)} x_j \geq |O|, \quad \forall i \in [n]. \tag{6.19}$$

Rewriting (6.19) in a matrix form, we get $(I + \mathcal{A}_{\mathcal{G}})x \geq (|O| + 1)\mathbf{1}$, where I is the identity matrix, $\mathcal{A}_{\mathcal{G}}$ is the adjacency matrix of $\mathcal{G}$, and the inequality is entrywise. Since we would like to assign the minimum number of capacities to the agents, we end up with the following integer program:

$$\begin{aligned} &\min \sum_{i=1}^{n} x_i \\ s.t \quad &(I + \mathcal{A}_{\mathcal{G}})x \geq (|O| + 1)\mathbf{1}, \quad x_i \in \mathbb{N}. \end{aligned} \tag{6.20}$$

In the above formulation, the capacities of the agents are restricted to be integer valued. However, we can relax the integer program (6.20) by assuming $x_i \in [1, \infty)$. Substituting $y = x - \mathbf{1}$ in the relaxed version of (6.20), we arrive at the following linear program:

$$\begin{aligned} &n + \min \sum_{i=1}^{n} y_i \\ s.t \quad &(I + \mathcal{A}_{\mathcal{G}})y \geq (|O| + 1)\mathbf{1} - \mathbf{d}, \quad y_i \geq 0, \end{aligned} \tag{6.21}$$

where $\mathbf{d} = (d_1, d_2, \dots, d_n)'$ is the column vector of the degrees of the graph $\mathcal{G}$. In fact, the entries of y denote the extra capacity that need to be added to each node in the network. Given a solution y^* of (6.21), let $\lceil y^* \rceil$ be a vector which is obtained by rounding off the entries of y^* to the closest integers greater than or equal to them. Note that since all the entries of A are non-negative, $\lceil y^* \rceil$ is a feasible solution for (6.21) and hence we have the following theorem.

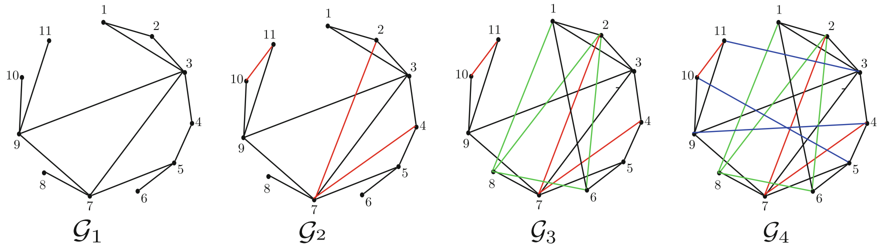


Fig. 6.10 Configuration of four networks $\mathcal{G}_1, \mathcal{G}_2, \mathcal{G}_3$ and $\mathcal{G}_4$. The new added edges from each network to the next one is identified with a different *color*

Theorem 6.15 *Consider an arbitrary network $\mathcal{G}$ of n nodes with adjacency matrix $\mathcal{A}_{\mathcal{G}}$ and a corresponding degree vector* **d**. *Then, the least best response dynamics will converge to an equilibrium of the CSR game in polynomial time $(n|O|)^2$ by adding $\lceil y_i^* \rceil$ extra cache for agent i, where y_i^* is a solution of the linear program* (6.21).

Proof The proof simply follows from the discussion preceding the theorem. Since the maximum cache size for each agent is $|O|$ we get $L \leq n|O|$. Hence using Theorems 6.3 and 6.14 we can find a NE in no more than $(n|O|)^2$ steps. □

6.11.1 A Numerical Example

In the following, we provide some numerical results of locating effective nodes in the network and adding extra capacities such that the least best response dynamics have faster convergence to an equilibrium. In Fig. 6.10 we have shown four different networks $\mathcal{G}_1, \mathcal{G}_2, \mathcal{G}_3$ and $\mathcal{G}_4$ in an increasing order of edge density. We consider the CSR game on these networks with $|O| = 5$ resources. In order to accelerate convergence to an equilibrium in each of these networks, we solve the linear program (6.21) for each of them and find the extra capacity y_i needed for each agent i. The numerical results are given in Table 6.1. Each column of the table corresponds to a different network and each pair $(y_i^*, \lceil y_i^* \rceil)$ denotes the optimal extra capacity and the up-rounded integer extra capacity for agent i. Also, the last two rows of Table 6.1 are related to the largest eigenvalue of each network (which is a measure of edge density) and the total extra capacity needed, normalized by the number of agents in the network. It can be seen that as long as the densities of the networks are allowed to increase, the need for extra cache will tend to zero.

Table 6.1 Solution of Linear Program (6.21) with $|O| = 5$ and for Networks $\mathcal{G}_1$, $\mathcal{G}_2$, $\mathcal{G}_3$ and $\mathcal{G}_4$

Solutions	$\mathcal{G}_1$	$\mathcal{G}_2$	$\mathcal{G}_3$	$\mathcal{G}_4$
y_1^*	0	0	0.38	0.32
y_2^*	0	1.13	0.82	1.18
y_3^*	3	1.87	0	0
y_4^*	0	0	0.19	0
y_5^*	4	4	0.78	0.5
y_6^*	0	0	0.30	0
y_7^*	4	4	1.49	0.5
y_8^*	0	0	0	0
y_9^*	4	3	3	0.5
y_{10}^*	0	0	0	0.5
y_{11}^*	0	0	0	0
—————-				
$\lambda_{\max}$	3.049	3.660	4.129	4.496
$\frac{\sum_{i=1}^{n} y_i^*}{n}$	1.3636	1.272	0.636	0.409

6.12 Discussion

In this section we discuss some future research directions related to the CSR (or LCSR) game as well as some new tools which can be leveraged to study properties of games under more general settings.

One question of interest is whether there exists a deterministic polynomial time approximation algorithm (instead of quasi-polynomial) which can approximate (or find) an NE of the CSR game within a constant factor, over general networks and arbitrary number of resources. We saw that dual approximation algorithm indeed runs in polynomial time over almost symmetric networks. In Sect. 6.5 we have shown that for any NE P^*, any optimal allocation P^o and any player i, we have $C_i(P^*) < \frac{1}{3}C_i(P^o)$. One may raise a similar question, as to whether there exists a constant α such that for any individual player i, any optimal allocation P^o, and any NE P^*, we have $C_i(P^o) \leq \alpha C_i(P^*)$. In fact, as a result of Theorems 6.5 and 6.1, one can see that this is the case when the underlying network has the tree structure. As another avenue for future research, a possibility is to study the dynamic version of the CSR game, in the spirit of what has been discussed in [10, 11].

Related to Sect. 6.9, one extension would be to explore duality under more general settings. Given a game which admits an ordinal potential function, one can define a partial ordering over the set $\mathcal{P}$ of all the states of the game such that for any two states $P, \hat{P} \in \mathcal{P}$ we have $P \leq \hat{P}$ if and only if starting from P, there exists a sequence of best responses which can change P to $\hat{P}$. By this definition, it must be clear that the maximal elements of such a partially ordered set (or simply poset) $\mathcal{P}$, are the NE points of the game. Adopting the same notions of chain and antichain from order theory (a chain in the poset $(\mathcal{P}, \leq)$ is a subset of its elements such that every two of them are comparable; an antichain is a subset of elements such that none of them are comparable), and using Dilworth's theorem [12], and its dual, Mirsky's theorem [13], one can see that the size of the longest chain (in our setting the most number of steps to reach an equilibrium using best response updates) equals to the minimum number of antichains which partition the poset $\mathcal{P}$. Therefore, the better the estimate we have about the size of the antichains in $(\mathcal{P}, \leq)$, the better we can bound the required number of updates to reach an equilibrium. As an example, in the CSR game we have $|\mathcal{P}| = |O|^n$, and all the allocation profiles which are missing one resource form an antichain of size at least $(|O| - 1)^n$ (since every better response of any player will bring the missing resource back to the profile). Therefore, such estimation of the size of the largest antichain will give us the existence of a chain of size at most $(1 + \frac{1}{|O|-1})^n$, i.e., the existence of a sequence of best response updates of length at most $(1 + \frac{1}{|O|-1})^n$ to an NE.

Finally, we want to point out that depending on the specific application, the best response dynamics are not always the best way to find an equilibrium. To handle such situations, one way is to define a function which can characterize the equilibrium points (it can be thought of as a weaker version of a potential function which is not always monotone along the trajectories of the better responses) and use some probabilistic techniques to prove the existence of some favorable states for such a function. As an example, let us reconsider the function $f(\cdot)$ which was defined in Lemma 6.6. In order to reach an NE, one could set f to be a function that needs to be maximized. Now the question is whether at every state of the game there exists at least one player whose update to some resource will result in an increase of the function f. Note that here f is not a potential function of the game, but it can be used to drive to equilibrium. In fact, a simple probabilistic argument shows that as long as the neighborhood of some player i is not saturated by more than $\sqrt{|O|}$ resources, then there exists a resource such that if player i updates to that specific resource, the value of function f strictly increases. However, such a function f performs well when players' neighborhoods are not saturated by more than $\sqrt{|O|}$ resources. In fact, for $M_i(t) > \sqrt{|O|}$ one can construct examples where f reaches some local maximum such that no individual's update can increase f even further. Therefore, exploring "weak potential" functions rather than potential functions for different types of problems and leveraging randomized algorithms for finding their global optimums would be another interesting research area (see, e.g., [14]).

6.13 Conclusion

In this chapter, we have studied the capacitated selfish replication (CSR) game over undirected networks. We have shown that optimal solutions of the CSR game constitute a subclass of its equilibrium points when the underlying network is a tree, and there exists a linear time algorithm to obtain one of these solutions. We introduced a class of dynamics known as the least best response dynamics which can find a pure Nash equilibrium when the number of resources is low or the underlying network is dense. We have provided a distributed quasi-polynomial time algorithm and its dual which can approximate any NE of the system as well as the optimal allocation profile within a constant factor. We have shown that lack of coordination among the players in the CSR game does not affect global optimality much. In other words, such games benefit from having a low price of anarchy, and can be utilized to model chaotic allocation systems where the players act in a completely selfish manner. Furthermore, we have formulated the equilibrium points and optimal solutions of the CSR game using integer linear programs, and provided an upper bound for the minimum social cost. We have introduced the LCSR game, a localized version of the CSR game, and elaborated on some of its properties. Using a network transformation, we have argued that all of the above results can be extended to cases where the players have different cache sizes. Finally, we have shown that increasing the capacity of the agents can improve the speed of convergence of the least best response dynamics to an equilibrium, and the location of effective nodes and the extra cache size can be found by solving a linear program.

References

1. R. Gopalakrishnan, D. Kanoulas, N. N. Karuturi, C. P. Rangan, R. Rajaraman, and R. Sundaram, "Cache me if you can: capacitated selfish replication games," in *LATIN 2012: Theoretical Informatics*. Springer, 2012, pp. 420–432.
2. I. Milchtaich, "Congestion games with player-specific payoff functions," *Games and Economic Behavior*, vol. 13, no. 1, pp. 111–124, 1996.
3. D. Monderer and L. S. Shapley, "Potential games," *Games and Economic Behavior*, vol. 14, no. 1, pp. 124–143, 1996.
4. T. Başar and G. J. Olsder, *Dynamic Noncooperative Game Theory*. SIAM, 1999, vol. 23.
5. B. Bollobás, *Random Graphs*. Cambridge University Press, 2001, vol. 73.
6. D. J. Watts and S. H. Strogatz, "Collective dynamics of small-world networks," *nature*, vol. 393, no. 6684, pp. 440–442, 1998.
7. S. R. Etesami and T. Başar, "Price of anarchy and an approximation algorithm for the binary-preference capacitated selfish replication game," *Automatica*, vol. 76, pp. 153–163, 2017.
8. S. R. Etesami and T. Başar, "An approximation algorithm and price of anarchy for the binary-preference capacitated selfish replication game," in *54th IEEE Conference on Decision and Control (CDC)*. IEEE, 2015, pp. 3568–3573.
9. S. R. Etesami and T. Başar, "Pure Nash equilibrium in a capacitated selfish resource allocation game," *IEEE Transactions on Control of Network Systems (to appear)*, 2017.
10. A. Fabrikant, A. Luthra, E. Maneva, C. H. Papadimitriou, and S. Shenker, "On a network creation game," in *Proceedings of the Twenty-Second Annual Symposium on Principles of Distributed Computing*. ACM, 2003, pp. 347–351.
11. B. Kawald and P. Lenzner, "On dynamics in selfish network creation," *arXiv preprint* arXiv:1212.4797, 2012.
12. R. P. Dilworth, "A decomposition theorem for partially ordered sets," *Annals of Mathematics*, pp. 161–166, 1950.
13. L. Mirsky, "A dual of Dilworth's decomposition theorem," *American Mathematical Monthly*, pp. 876–877, 1971.

14. S. N. Ahmadyan, S. R. Etesami, and H. V. Poor, “A random tree search algorithm for Nash equilibrium in capacitated selfish replication games,” in *55th Conference on Decision and Control (CDC)*. IEEE, 2016, pp. 4439–4444.

Chapter 7
Diffusion Games over Social Networks

One of the widely studied models in social networks is the diffusion model, where the goal is to propagate a certain type of product or behavior in a desired way through the network. Examples include online advertising for companies' products, propagation of rumors and computer viruses, and epidemics. One of the challenges within this context is to understand how to control the diffusion process in a proper way by possibly targeting and investing on the most influential entities in the underlying social network. This problem becomes even more complicated when there are several parties who plan to maximize their benefits by diffusing their own products. This brings up the notion of competitive diffusion over the social networks, where game theoretic tools seem quite appealing and effective.

In this chapter we study the competitive diffusion game as was introduced earlier in [1] and obtain several results regarding its Nash equilibria [2, 3]. For sake of simplicity, and without much loss of conceptual generality, we state the model when there are only two players in the game; however, the model and analysis can readily be extended to the case when there are more than two players.

7.1 The Diffusion Game Model

Following the formulation in [1], we consider here a network $\mathcal{G}$ of n nodes and two players (types) A and B. Initially at time $t = 0$, each player decides to choose a subset of nodes in the network and place his own seeds. After that, a discrete time diffusion process unfolds among uninfected nodes as follows:

- If at some time step t an uninfected node is neighbor to infected nodes of only one type (A or B), it will adopt that type at the next time step $t + 1$.

S.R. Etesami, *Potential-Based Analysis of Social, Communication, and Distributed Networks*, Springer Theses, DOI 10.1007/978-3-319-54289-8_7

- If an uninfected node is connected to nodes of both types at some time step t, it will change to a gray node at the next time step $t + 1$ and does not adopt any type afterward.

This process continues until no new adoption happens. Finally, the utility of each player will be the total number of infected nodes of its own type at the end of the process. Moreover, if both players place their seeds on the same node, that node will change to gray. We want to emphasize the fact that when a node changes to gray, not only will it not adopt any type at the next time step, but also it may block the streams of diffusion to other uninfected nodes. We will see later that the existence of gray nodes in the evolution of the process can make any prediction process about the outcome of the diffusion process much more complicated.

Remark 19 The diffusion process as defined above is a particular case of progressive diffusion processes, where the state of the nodes does not change after adoption. This is in contrast to other types of processes known as non-progressive processes [4].

Remark 20 For the case of $k > 2$ players, one can define the same process as above such that an uninfected node will adopt type i at time $t + 1$ if and only if type i is the only existing type among its neighbors at time step t.

As mentioned earlier, it has been shown in [1, 5, 6] that the competitive diffusion game may or may not admit pure-strategy Nash equilibria depending on the topology of the network $\mathcal{G}$, and the number of players. Moreover, it has been shown in [7] that if the underlying graph $\mathcal{G}$ has a tree structure, the diffusion game with two players admits a pure-strategy Nash equilibrium. In fact, for the case of three or more players even the tree structure may not lead to a pure Nash equilibrium. In the next sections we will study some of the properties of the diffusion process over some specific as well as general networks, and in particular obtain some conditions which are necessary for the existence of at least one equilibrium.

7.2 The Diffusion Game over Specific Networks

In this section, we consider the 2-player diffusion game with a single seed placement and study the existence of a pure-strategy Nash equilibrium for two special but well-studied classes of networks, namely the *lattice* and the *hypercube*. Such an analysis sheds light on the problem under more general settings, which is the topic of the next section.

Definition 7.1 An $m \times n$ lattice is a graph $L_{m\times n}$ with vertex set $V = \{(x, y) \in \mathbb{Z}^2 : 0 \leq x \leq m, 0 \leq y \leq n\}$ such that each node is adjacent to those whose Euclidean distanced is 1 from it. A k dimensional hypercube is a graph Q_k with vertex set $\{0, 1\}^k$ such that two k-tuples are adjacent if and only if they differ in exactly one position.

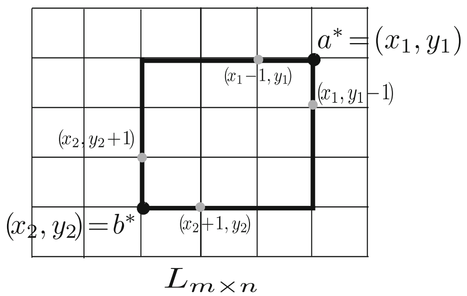


Fig. 7.1 Illustration of lattice in Proposition 7.1

Proposition 7.1 *For the 2-player diffusion game over the lattice $L_{m\times n}$, $m, n \in Z^+$, a profile (a^*, b^*) is a Nash equilibrium if and only if a^* and b^* are adjacent nodes in the most centric square or edges of the $L_{m\times n}$.*

Proof Let us assume that $a^* = (x_1, y_1)$ and $b^* = (x_2, y_2)$ are two nonadjacent nodes in $L_{m\times n}$ which form a pure-strategy Nash equilibrium. Without any loss of generality, and if necessary by rotating the lattice $L_{m\times n}$ and relabeling the types, we can assume that $x_1 > x_2$ and $y_1 \geq y_2$. This situation has been shown in the Fig. 7.1. Now, it is not hard to see that at least one of the following cases will happen:

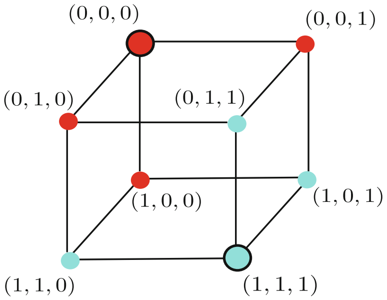
- Player B can strictly increase her utility by deviating to either $(x_1 - 1, y_1)$ or $(x_1, y_1 - 1)$.
- Player A can strictly increase her utility by deviating to either $(x_2 + 1, y_2)$ or $(x_2, y_2 + 1)$.

Therefore, in any of these cases, it means that (a^*, b^*) cannot be an equilibrium. Moreover, if two adjacent nodes are not in the most centric part of the lattice, it can be seen that one of the players can always increase her utility by deviating to another neighbor of the other player while moving closer to the center of the lattice. This shows that if a profile is a Nash equilibrium it must be two adjacent nodes in the most centric square of the lattice $L_{m\times n}$. Finally, using the same line of argument as above, it is not hard to see that in every such profile each of the players will obtain the maximum utility that she can get, given that the position of the other player is fixed. This shows that indeed any adjacent profile in the most centric square is a Nash equilibrium. □

Proposition 7.2 *A profile (a^*, b^*) is a Nash equilibrium of the 2-player diffusion game over the hypercube Q_k if and only if the graphical distance between a^* and b^* is an odd number, or equivalently $(a^* \oplus b^*)'\mathbf{1}_k = 1$ mod 2.*

Proof First, we note that if $(a^* \oplus b^*)'\mathbf{1}_k = 1 \bmod 2$, then $U_A(a^*, b^*) = U_B(a^*, b^*) = 2^{k-1}$, where $U_A(a^*, b^*)$ and $U_B(a^*, b^*)$ denote, respectively, the utilities of players A and B given that their initial seeds are at a^* and b^*. In this case, there exists no vertex v which has equal graphical distance to both a^* and b^*; otherwise, if $d(a^*, v) = d(b^*, v)$, it means that $(a^* \oplus v)'\mathbf{1} = (b^* \oplus v)'\mathbf{1}$. Therefore,

Fig. 7.2 A Nash equilibrium of the diffusion game over 3-dimensional hypercube. The *circled nodes* denote the initial seeds and the figure illustrates the final state of the game



$(a^* \oplus b^*)'\mathbf{1} = (a^* \oplus v)'\mathbf{1} + (b^* \oplus v)'\mathbf{1} = 0 \mod 2$, which is in contradiction with the assumption. Therefore, using Lemma 7.1, every node of Q_k must adopt either type A or B. Finally, for every node v such that $d(a^*, v) < d(b^*, v)$, one can assign a unique node $u = v \oplus (a^* \oplus b^*)$, such that $d(b^*, u) < d(a^*, u)$. This one-to-one bijection shows that $U_A(a^*, b^*) = U_B(a^*, b^*) = \frac{|V(Q_k)|}{2} = 2^{k-1}$.

To show that every such profile is indeed a pure Nash equilibrium, we show that for every arbitrary profile, the maximum utility that each player can gain is at most 2^{k-1}. We show it by induction on k. For $k = 1$, the result is trivial. Assuming that the statement is true for $k - 1$, let (a^*, b^*) be an arbitrary pair of nodes in Q_k. We consider two cases:

- $a^* \oplus b^* = \mathbf{1}_k$. In this case a^* and b^* are binary complementary of each other (by a simple translation of each node by the binary vector a^* and without any loss of generality we may assume $a^* = 0$ and $b^* = \mathbf{1}_k$). Therefore, by symmetry of Q_k, it is not hard to see that $U_A(a^*, b^*) = U_B(a^*, b^*)$. Since the total utility is at most 2^k, thus we have $U_A(a^*, b^*) = U_B(a^*, b^*) \leq 2^{k-1}$. Such a situation for the case of $k = 3$ is illustrated in Fig. 7.2.
- $a^* \oplus b^* \neq \mathbf{1}_k$. In this case, there exists at least one coordinate such that a^* and b^* agree on it. Without any loss of generality, let us assume $a_i^* = b_i^*$ and let $Q = \{x \in V(Q_k) : x_i = a_i^*\}$ and $Q^c = V(Q_k) \setminus Q$. Therefore, it is not hard to see that the induced subgraphs by vertices Q and Q^c are hypercubes of dimension $k - 1$ which are connected to each other through a perfect matching (a set of disjoint edges which connect all the nodes in pairs). Furthermore, a^* and b^* are both located in Q. Therefore, by the induction hypothesis, the utility of the players on Q, i.e., $U_A^Q(a^*, b^*)$, $U_B^Q(a^*, b^*)$ is at most 2^{k-2}. Moreover, since Q and Q^c are connected through a perfect matching and there is only one step delay in the diffusion process between Q and Q^c, hence the adoption of vertices in Q^c is exactly the same as those in Q. Therefore, we have $U_A(a^*, b^*) = 2U_A^Q(a^*, b^*) \leq 2 \times 2^{k-2} = 2^{k-1}$, and similarly $U_B(a^*, b^*) \leq 2^{k-1}$.

Overall, we have shown that a profile (a^*, b^*) in Q_k is a Nash equilibrium if and only if $U_A(a^*, b^*) = U_B(a^*, b^*) = 2^{k-1}$, which happens if and only if a^* and b^* have an odd graphical distance from each other, or equivalently $(a^* \oplus b^*)'\mathbf{1}_k = 1 \mod 2$. □

7.3 Preliminary Results on the Diffusion Game Model

In this section, we first characterize the final state of the diffusion process based on relative distances of the players' initial seed placements on the network. In the next section and by using this characterization we establish a hardness result for the decision process on the existence of a pure-strategy Nash equilibrium in the diffusion game.

Lemma 7.1 *Let $\mathcal{N}_A$ and $\mathcal{N}_B$ denote the set of nodes which adopt, respectively, types A and B at the end of the process for a particular initial selection of seeds $a, b \in V$. Moreover, let $\mathcal{N}$ be the set of gray or uninfected (white) nodes by the end of the process. Then,*

$$\begin{aligned} \mathcal{N} &\subseteq \{i : d_{\mathcal{G}}(a,i) = d_{\mathcal{G}}(b,i)\}, \\ \{i : d_{\mathcal{G}}(a,i) < d_{\mathcal{G}}(b,i)\} \subseteq \mathcal{N}_A &\subseteq \{i : d_{\mathcal{G}}(a,i) \leq d_{\mathcal{G}}(b,i)\}, \\ \{i : d_{\mathcal{G}}(b,i) < d_{\mathcal{G}}(a,i)\} \subseteq \mathcal{N}_B &\subseteq \{i : d_{\mathcal{G}}(b,i) \leq d_{\mathcal{G}}(a,i)\}. \end{aligned} \tag{7.1}$$

Proof The proof is by induction on the time step $t = 0, 1, \ldots$. At time $t = 0$ there is no gray node and the result follows. At time $t = 1$, if there exist some gray nodes, it requires the gray nodes to be neighbors of both i_A and i_B, as otherwise, they will adopt B or A, respectively. Therefore, for all gray nodes j which are born at time $t = 1$ we must have $d_{\mathcal{G}}(i_A, j) = d_{\mathcal{G}}(i_B, j) = 1$. Now, assume that the statement of the Lemma is true for all the gray nodes such that they are born at steps $t \leq k$, and let T_k be the collection of all these nodes. Consider a gray node ℓ which is born at time $t = k + 1$. Note that if there is no such node, there is nothing to show and we can go one step forward. Assume P_1 and P_2 are two different shortest paths from i_A and i_B to ℓ, respectively. We consider the following cases:

1. P_1 and P_2 do not have any other gray node except ℓ.
2. P_1 (or similarly P_2) has at least one more gray node other than ℓ, but P_2 (or similarly P_1) does not have any other gray node except ℓ.
3. P_1 and P_2 both have at least one gray node other than ℓ.

In the first case, we note that since ℓ has changed to gray at time $t = k + 1$ and since the shortest paths P_1 and P_2 have no gray nodes inside, we thus conclude that $d_{\mathcal{G}}(i_A, \ell) = d_{\mathcal{G}}(i_B, \ell) = k + 1$, as otherwise ℓ will adopt either A or B. Moreover, after k time steps, all the internal nodes of P_1 and P_2 will adopt A and B, respectively.

In the second case, let z be the first internal gray node in P_1 which is born by running the process from time $t = 0$ to time $t = k$. Since P_1 is the shortest path between i_A and ℓ, and also z is located on this path, we observe that $P_1(i_A \to \ell)$ is also the shortest path between i_1 and z. Since by the definition of z there is no other gray node on $P_1(i_A \to z)$, we thus conclude that z is born at step $t_0 = |P_1(i_A \to z)| < |P_1(i_A \to \ell)|$ and hence $z \in T_k$. Moreover, since P_1 is the shortest path between the initial seed i_A and ℓ, and since ℓ changes to gray at step $k + 1$, it means that ℓ does not change to gray for the first $|P_1(i_A \to \ell)| - 1$ steps. Thus, $|P_1(i_A \to \ell)| \leq k + 1$.

Putting these inequalities together, we get $t_0 \leq k$ and by the induction hypothesis we have $d_{\mathcal{G}}(i_A, z) = d_{\mathcal{G}}(i_B, z)$. Now we can write

$$d_{\mathcal{G}}(i_A, \ell) = d_{\mathcal{G}}(i_A, z) + d_{\mathcal{G}}(z, \ell) = d_{\mathcal{G}}(i_B, z) + d_{\mathcal{G}}(z, \ell) \geq d_{\mathcal{G}}(i_B, \ell). \tag{7.2}$$

We claim that equality must hold in (7.2). Otherwise, if $d_{\mathcal{G}}(i_A, \ell) > d_{\mathcal{G}}(i_B, \ell)$, then since P_2 does not have any gray node other than ℓ, it requires that after $d_{\mathcal{G}}(i_B, \ell) - 1$ steps all the nodes in P_2 other than ℓ adopt B. Therefore, at time step $d_{\mathcal{G}}(i_B, \ell) - 1$ node ℓ has at least one neighbor who has adopted B. On the other side, since $d_{\mathcal{G}}(i_A, \ell) > d_{\mathcal{G}}(i_B, \ell)$, node ℓ has no neighbor of type A. This means that at time step $t = d_{\mathcal{G}}(i_B, \ell)$ agent ℓ will adopt B. This is in contradiction with the fact that ℓ has a gray color and thus we have $d_{\mathcal{G}}(i_A, \ell) = d_{\mathcal{G}}(i_B, \ell)$. Finally in the last case, by repeating the same argument given in the second case for the seed nodes i_A and i_B, we get $d_{\mathcal{G}}(i_A, \ell) \geq d_{\mathcal{G}}(i_B, \ell)$ and $d_{\mathcal{G}}(i_B, \ell) \geq d_{\mathcal{G}}(i_A, \ell)$, which gives us $d_{\mathcal{G}}(i_A, \ell) = d_{\mathcal{G}}(i_B, \ell)$.

Now, let us assume that ℓ' is a node which remains uninfected (white) at the end of the process. It means that there is no path of types either A or B which connects the seed nodes i_A or i_B to ℓ'. Without loss of generality assume that $d_{\mathcal{G}}(i_A, \ell') < d_{\mathcal{G}}(i_B, \ell')$ and denote one of the shortest paths between i_A and ℓ' by P_1'. This path must have at least one gray node which we denote by z'. We can write

$$d_{\mathcal{G}}(i_A, \ell') = d_{\mathcal{G}}(i_A, z') + d_{\mathcal{G}}(z', \ell') = d_{\mathcal{G}}(i_B, z') + d_{\mathcal{G}}(z', \ell') \geq d_{\mathcal{G}}(i_B, \ell').$$

This contradiction shows that $d_{\mathcal{G}}(i_A, \ell') = d_{\mathcal{G}}(i_B, \ell')$. Gathering all the above results, we get

$$\mathcal{N} \subseteq \{i : d_{\mathcal{G}}(i_A, i) = d_{\mathcal{G}}(i_B, i)\}. \tag{7.3}$$

To complete the proof, we only need to show one of the relations in (7.1), as the proof for the other one would be the same. Assume that for some i^* we have $d_{\mathcal{G}}(i_A, i^*) < d_{\mathcal{G}}(i_B, i^*)$. Using (7.3) we know that i^* must adopt either A or B at the end of the process. Let P_1^* be one of the shortest paths between i_A and i^*. Moreover, assume i^* will adopt B at the end of the process. The only reason why this happens is that there is a gray node on P_1^* such that it blocks the diffusion of type A from i_A to i^* as otherwise type A will reach i^* earlier than type B. Let this gray node be z^*. Using (7.3) we know that $d_{\mathcal{G}}(i_A, z^*) = d_{\mathcal{G}}(i_B, z^*)$. Thus,

$$d_{\mathcal{G}}(i_A, i^*) = d_{\mathcal{G}}(i_A, z^*) + d_{\mathcal{G}}(z^*, i^*) = d_{\mathcal{G}}(i_B, z^*) + d_{\mathcal{G}}(z^*, i^*) \geq d_{\mathcal{G}}(i_B, i^*).$$

This contradiction shows that i^* will adopt A and hence $\{i : d_{\mathcal{G}}(i_A, i) < d_{\mathcal{G}}(i_B, i)\} \subseteq \mathcal{N}_A$. This completes the proof. □

Note that in Lemma 7.1 we assumed that each player can only place one seed in the network as its initial placement. However, the result can be easily generalized to the case when each player (for example player A) is allowed to choose a set of nodes

$S_A \subset V$ as its initial seed placements. In this case we just need to replace $d_{\mathcal{G}}(S_A, x)$ instead of $d_{\mathcal{G}}(a, x)$ in the statement of Lemma 7.1 and all the other results carry over naturally. Moreover, a similar result can be proved when there are more than two players in the game.

7.4 NP-Hardness of Making a Decision on the Existence of Nash Equilibrium

Departing from the 2-player framework of Sect. 7.2, we are now considering a game with $m \geq 2$ players. As it was shown before in [1, 6], one can always construct networks with diameter greater than or equal to 2 such that the diffusion game over such networks does not admit any pure-strategy Nash equilibrium. In fact, by a closer look at Lemma 7.1, one can see that there is some similarity between Voronoi games [8] and competitive diffusion games. Note, however, that in the competitive diffusion game a diffusion process unfolds while there is no notion of diffusion in Voronoi games. However, since at the end of the process both games demonstrate behavior close to each other, it seems natural to compare the complexity of Nash equilibria in these two games. In fact, in the following we show that the decision on the existence of Nash equilibrium in a diffusion game is NP-hard. Toward that goal, we first prove some relevant results and modify the configuration of the diffusion game to make a connection with Voronoi games. Borrowing some of the existing results from the Voronoi games literature, we prove the NP-hardness of verification of existence of Nash equilibrium in the diffusion game. We prove it by reduction from the 3-partitioning problem which is shown to be an NP-complete problem [9]. In the 3-partitioning problem, we are given integers $\alpha_1, \alpha_2, \ldots, \alpha_{3m}$ and a β such that $\frac{\beta}{4} < \alpha_i < \frac{\beta}{2}$ for every $1 \leq i \leq 3m$, $\sum_{i=1}^{3m} \alpha_i = m\beta$ and have to partition them into disjoint sets $P_1, \ldots, P_m \subseteq \{1, 2, \ldots, 3m\}$ such that for every $1 \leq j \leq m$ we have $\sum_{i \in P_j} \alpha_i = \beta$.

First, we briefly describe the stages that we will go through toward proving the NP-hardness of making a decision on the existence of a Nash equilibrium. Given an arbitrary network $\mathcal{G}$, by adding extra nodes and edges properly we expand this network into a new network $\bar{\mathcal{G}}$ such that we make sure that if there exists a Nash equilibrium in the original network, it must lie within a confined subset of nodes in the extended network $\bar{\mathcal{G}}$ (Lemma 7.2). This allows us to confine our attention to only a specific subset of nodes in the extended network in order to search for a Nash equilibrium. Following this, we construct a new network (Theorem 7.3) such that any Nash equilibrium of the game is equivalent to a solution of the 3-partitioning problem which is known to be an NP-complete problem. This establishes the NP-hardness of arriving at a decision on the existence of a Nash equilibrium in the diffusion game. We begin a formal proof of the result by stating the following lemma.

Lemma 7.2 *Let T be a subset of $V(\mathcal{G})$. Then, there exists an extended graph $\bar{\mathcal{G}}$ such that there is a bijection between the Nash equilibria in $\mathcal{G}$ where the seeds (actions of the players) are restricted to T, and the unrestricted Nash equilibria in $\bar{\mathcal{G}}$.*

Proof Consider the graph $\bar{\mathcal{G}}$ depicted in Fig. 7.3, which is constructed using $\mathcal{G}$ by adding $|T|n$ new nodes and $n\frac{|T|(|T|+1)}{2}$ new edges. Note that $|T|$ denotes the number of the nodes in T and $n = 2|V(\mathcal{G})| + 1$ is a positive integer. It is important to note here that although $n > |V(\mathcal{G})|$, it is polynomially bounded above by $|V(\mathcal{G})|$. With each node $i \in T$ we associate a set of n new nodes $C_i = \{c_{i1}, c_{i2}, \ldots, c_{in}\}$ and we connect all of them to node i. Furthermore, for $j = 1, \ldots, n$, we connect all the nodes $c_{1j}, \ldots, c_{|T|j}$ to each other. In other words, nodes $\{c_{1j}, \ldots, c_{|T|j}\}$ form a clique for each $j = 1, \ldots, n$.

Now assume that at least one player puts his node seed in $k \in T$. We refer to this player as the *first player* and denote its type by A. We claim that all the other players must play in T as well. To prove this, suppose that another player which we refer to as the *second player* with corresponding type B chooses node $\ell \in T, \ell \neq k$. In this case he will earn at least n due to winning all the nodes in C_ℓ. Now let us assume that the second player plays in the bottom part of the graph (Fig. 7.3), i.e. $L = \cup_{i=1}^{|T|} C_i$. We consider two cases:

1. He plays in C_k and without any loss of generality and by symmetry, we may assume that he plays in c_{k1}.
2. He plays in C_ℓ for some $\ell \neq k$ such as $c_{\ell 1}$.

In the first case and after the first step of diffusion, all the elements $c_{11}, c_{21}, \ldots, c_{|T|1}$ will adopt type of the second player, i.e. B (because there is a direct link between them and c_{k1}), and all the elements $c_{k2}, c_{k3}, \ldots, c_{kn}$ will adapt type of the first player, i.e. A. At the second time step, the second player can adopt all the elements of $T\backslash\{k\}$ in the best case. On the other hand, all the elements of $L\backslash\{c_{11}, c_{21}, \ldots, c_{|T|1}\}$ will change to A. Therefore, in this case the second player can gain at most $|T| + |V(\mathcal{G})| \leq 2|V(\mathcal{G})| < n$.

In the second case and after the first step of diffusion, the second player can adopt only $\{c_{11}, c_{21}, \ldots, c_{|T|1}, \ell\} \setminus \{c_{k1}\}$ while the first player will adopt at least $\{c_{k2}, \ldots, c_{kn}\}$, (note that node c_{k1} will change to gray). However, at the second time step all the nodes in $C_\ell\backslash\{c_{\ell 1}\}$ and also in $L\backslash\big\{C_k \cup C_\ell \cup \{c_{11}, c_{21}, \ldots, c_{|T|1}\}\big\}$ will change to type A. Therefore, in this case, the second player gains at most $|T| + |V(\mathcal{G})| \leq 2|V(\mathcal{G})| < n$.

Furthermore, if the second player places his seed at a node in $V(\mathcal{G})\backslash T$, then in the best scenario it will take at least two steps for the seed to be diffused to nodes of L. On the other hand, type A can be diffused through every node of L in no more than 2 steps. Thus all the nodes in L either adopt A or they change to gray and hence, in this case the second player cannot earn more than $V(\mathcal{G}) - 1 < n$. From the above discussion it should be clear that in either of the above cases, if a player is playing in $L \cup V(\mathcal{G})\backslash T$ he can always gain more by deviating to the set T. Thus in each equilibrium players must put their seeds in T.

Finally suppose that $\bar{Q}$ is a Nash equilibrium profile in $\bar{\mathcal{G}}$. By the above argument we know that all the players must play in T and thus, each of these players gains

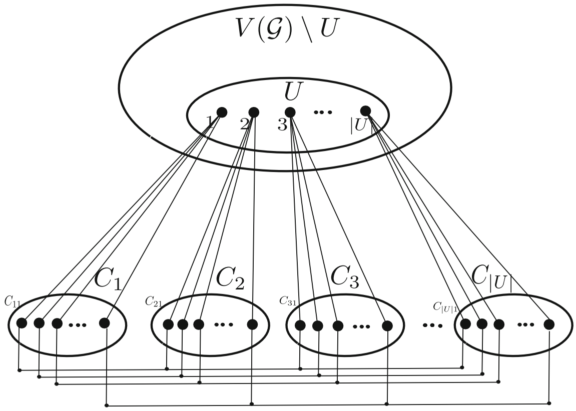


Fig. 7.3 Extension of graph $\mathcal{G}$ to $\bar{\mathcal{G}}$

exactly n from the set L. Therefore, the utility of players is equal to the utility that they would gain by playing on $\mathcal{G}$ plus n. This shows that $\bar{Q}$ must be an equilibrium for $\mathcal{G}$ when the strategies of players are restricted to T. Similarly, if Q is an equilibrium of $\mathcal{G}$ when the strategies of players are restricted to T, it is also an equilibrium for $\bar{\mathcal{G}}$ as we know all the equilibria seeds of players (if there is any) must be in T. This shows that the set of equilibria of $\bar{\mathcal{G}}$ is equivalent to the set of equilibria of $\mathcal{G}$ with the restricted strategy set T. □

Theorem 7.3 *Given a graph $\mathcal{G}$ and $m \geq 2$ players, the decision process on the existence of pure-strategy Nash equilibrium for the diffusion game on $\mathcal{G}$ is NP-hard.*

Proof Assume that integers $\alpha_1, \alpha_2, \ldots, \alpha_{3m}$ and $\beta > 3$ are given such that $\frac{\beta}{4} < \alpha_i < \frac{\beta}{2}$ for every $1 \leq i \leq 3m$, $\sum_{i=1}^{3m} \alpha_i = m\beta$. Moreover, let $c = \binom{3m}{3}$ and choose an integer d such that $\frac{(\beta-1)c}{4} < d < \frac{\beta c}{4}$. Consider the graph depicted in Fig. 7.4, where the set T is defined to be the set of vertices whose induced subgraph $\mathcal{G}[T]$ is illustrated in the dashed-line area. In fact, such a graph is composed of three parts:

- The right graph: This is a graph of $9d$ nodes, composed of 9 stars of size d where the centers of the stars are connected as it is shown in Fig. 7.4. It has been shown in [6] that this graph does not admit any pure-strategy Nash equilibrium when there are two agents. In fact, if there is only one player on this graph, he will gain $9d$ and if there are two players on it, one of them can always deviate to gain at least $4d$.
- The middle graph: This graph is simply a clique of size $\binom{3m}{3}$ where the nodes are labeled by $u_{i,j,k}$ for all possible triples $\{i, j, k\}$, $i, j, k \in [3m]$.
- The left graph: This graph is composed of $3m$ independent sets (i.e., there exists no edge between vertices of each set) $\mathcal{I}_1, \ldots, \mathcal{I}_{3m}$ of sizes $c\alpha_1, \ldots, c\alpha_{3m}$, respectively, such that all the nodes in the independent sets $\mathcal{I}_i$, $\mathcal{I}_j$, and $\mathcal{I}_k$ are connected to node $u_{i,j,k}$ in the middle graph.

Calling the graph in Fig. 7.4 $\mathcal{G}$, setting T to be the set of vertices of the middle and right side graphs, and applying Lemma 7.2 to construct $\bar{\mathcal{G}}$, we can see that any Nash

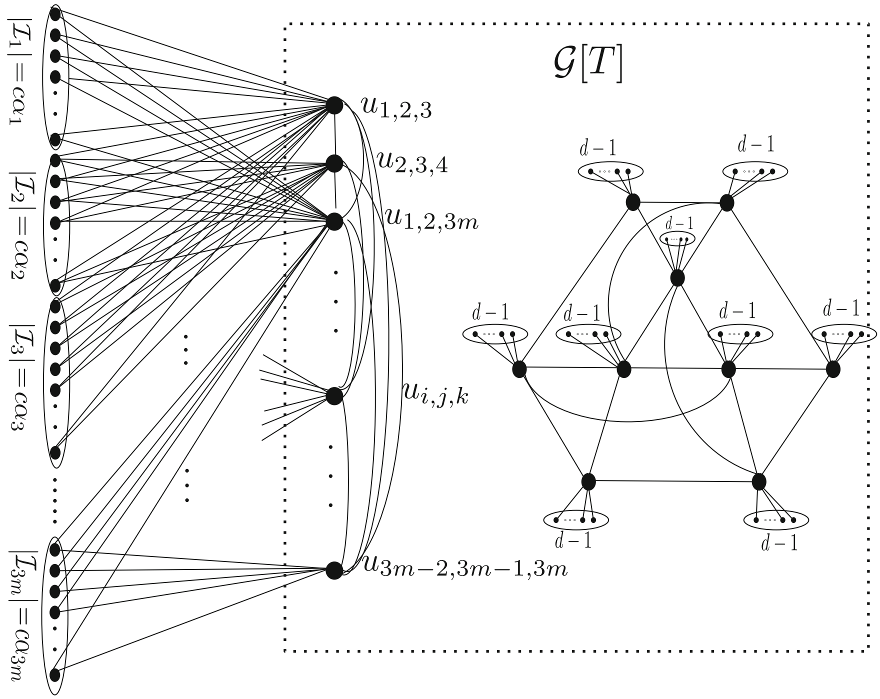


Fig. 7.4 Graph construction of Theorem 7.3

equilibrium of $\bar{\mathcal{G}}$ is an equilibrium of $\mathcal{G}$ when the strategies of players are restricted to T. We claim that Q is an equilibrium for $\mathcal{G}$ with the restricted strategy set T (and equivalently an equilibrium for unrestricted $\bar{\mathcal{G}}$) if and only if there is a 3-partitioning of $\{\alpha_i\}_{i=1}^{3m}$.

Assume that there is a solution $P_1, \ldots, P_m$ to the 3-partition. For every $1 \leq q \leq m$, if $P_q = \{i, j, k\}$, then player q is assigned to $u_{i,j,k}$. Let us also assume that player $m+1$ is assigned to one of the nodes in the rightmost part of the graph, which makes his utility to be $9d$. If player $m+1$ moves to a vertex $u_{i,j,k}$ his utility will be $1 < 9d$, because all the other m players already covered all the $\sum_{\ell=1}^{3m} c a_\ell$ nodes in the leftmost side of the graph and his movement will not result in any additional payoff for him except producing some gray nodes. Now if one of the players $1 \leq q \leq m$ moves from vertex $u_{i,j,k}$ to one of the nodes in the right part of the graph, then his gain can be at most $4d$ which by the selection of d would be less than what he was getting before (βc). Finally, if player q or equivalently node $u_{i,j,k}$ moves to another node $u_{i',j',k'}$ for some $\{i', j', k'\} \neq \{i, j, k\}$, then since P_q was part of 3-partitioning before, it means that his payoff after deviating will be at most $c \max\{\alpha_i + \alpha_j, \alpha_i + \alpha_k, \alpha_j + \alpha_k\} < c\beta$. Moreover, by Lemma 7.2 no player at equilibrium will be out of T and hence the proposed profile using the 3-partitioning is an equilibrium.

Now let us suppose that there exists a Nash equilibrium for $\bar{\mathcal{G}}$. We show that it corresponds to a solution of 3-partitioning. First we note that there cannot be two players in the rightmost part of the graph; otherwise, it is not an equilibrium by [6]. Moreover, if there are 3 players or more, one of them can gain at most $3d$. Since in this case there are at most $m-2$ players in the middle part, we can find a set $\{i', j', k'\}$ such that the corresponding set of all the other players does not have any intersection with it. Therefore, if a player with the least gain (at most $3d$) from the right side deviates to $u_{i',j',k'}$ in the middle part, he will gain at least $\frac{3\beta c}{4}$ which is greater than $3d$. Thus the rightmost part can have either one player or nothing. However since $9d > \frac{3\beta c}{2}$, at least one player would want to move to the rightmost part of the graph if there is no other player there. Therefore, the rightmost part of the graph has exactly one player. Thus, the rest of the m players must not only play in T (because of Lemma 7.2) but also they must form a partition. Otherwise one of them can move to an appropriate vertex of the middle graph and increase his utility. Finally, in this partitioning, each player must gain exactly βc, because if this is not true and one of the players, namely $u_{i,j,k}$, gets less than βc, he will gain at most $(\beta - 1)c$ (note that c is a rescaling factor) and thus, he can always move to the rightmost side of the graph and gain $4d > (\beta - 1)c$. Thus this partitioning must be a 3-partitioning. This proves the equivalence of the existence of Nash equilibrium in $\bar{\mathcal{G}}$ and existence of 3-partitioning for the set $\{\alpha_i\}_{i=1}^{3m}$. □

7.5 Necessary Conditions for Nash Equilibrium in the Diffusion Game

In this section, we consider the 2-player diffusion game with a single seed placement, present some necessary conditions for existence of pure-strategy Nash equilibrium, and discuss its connection with the network structure. Here, it is worth noting that although the results of this section provide some necessary conditions for a given profile to be a Nash equilibrium, in general they do not guarantee the existence of a Nash equilibrium, that is, they are not sufficient. We start with the following theorem which is for the case of two players; however, it can be extended quite naturally to the case of an arbitrary (but finite) number of players.

Theorem 7.4 *Suppose that $(a^*, b^*) \in V \times V$ is an equilibrium profile for the diffusion game. Then,*

$$\lceil \frac{n-1}{d(a^*)} \rceil \leq U_B(a^*, b^*), \quad \lceil \frac{n-1}{d(b^*)} \rceil \leq U_A(a^*, b^*),$$

where $U_A(a^*, b^*)$ and $U_B(a^*, b^*)$ denote the utilities of players A and B given the initial seed placement at (a^*, b^*), and $d(a^*)$ and $d(b^*)$ denote, respectively, the degrees of nodes a^* and b^* in the network $\mathcal{G}$.

Proof Assume that players A and B place their seeds at nodes a^* and b^* and receive payoffs of $U_A(a^*, b^*)$ and $U_B(a^*, b^*)$, respectively. We claim that there exists a neighbor of a^* where player b^* can gain at least $\lceil \frac{n-1}{d(a^*)} \rceil$ by deviating to it. Toward showing this, let us also denote all the neighbors of a^* by $v_1, v_2, \ldots, v_{d(a^*)}$. Let us denote the nodes that adopt B for the initial seed allocation (a^*, v_i) by S_i, $i = 1, 2, \ldots, d(a^*)$. Then, we have $\cup_{i=1}^{d(a^*)} S_i = V \backslash \{a^*\}$. In fact, for every $v \in V\backslash\{a^*\}$, the shortest path from v to a^* must pass through at least one of the neighbors of a^* such as v_ℓ. This means that $d_{\mathcal{G}}(v, v_\ell) < d_{\mathcal{G}}(v, a^*)$ and using Lemma 7.1 we can see that $v \in S_\ell$. Therefore we have $n - 1 = | \cup_{i=1}^{d(a^*)} S_i| \leq \sum_{i=1}^{d(a^*)} |S_i|$ and this means that there exists at least one v_{i^*} such that $|S_{i^*}| \geq \lceil \frac{|V\backslash\{a^*\}|}{d(a^*)} \rceil = \lceil \frac{n-1}{d(a^*)} \rceil$. Since we assumed that (a^*, b^*) is an equilibrium, player b^* cannot gain more by deviating to v_{i^*}. This means that $\lceil \frac{n-1}{d(a^*)} \rceil \leq U_B(a^*, b^*)$ and using the same argument for the other player we get $\lceil \frac{n-1}{d(b)} \rceil \leq U_A(a^*, b^*)$. □

Note that the results in Theorem 7.4 can be improved by noting that the inequality $| \cup_{i=1}^{d(a^*)} S_i| \leq \sum_{i=1}^{d(a^*)} |S_i|$ can be strict and there are nodes which might be counted in different sets of S_i. In fact, it is not hard to see that if a node v belongs to two of these sets such as S_j and S_k, v must be in an even cycle emanating a^* and including the nodes v_j and v_k. In such a case, to every cycle of even length which includes a^* and does not contain another smaller cycle, one can associate a node which is counted twice in two different sets. We call such cycles *simple even cycles emanating from* a^*. Therefore, we can write

$$n - 1 = | \cup_{i=1}^{d(a^*)} S_i| \leq \sum_{i=1}^{d(a^*)} |S_i| - \{\text{Simple even cycles emanating from } a^*\}|,$$

and therefore the bound in Theorem 7.4 will change to

$$\left\lceil \frac{n-1+|\{\text{Simple even cycles emanating from } a^*\}|}{d(a^*)} \right\rceil \leq U_B(a^*, b^*).$$

Next, we consider the following two definitions from graph theory.

Definition 7.2 An edge (a vertex) of a connected graph $\mathcal{G}$ is a *cut-edge (cut-vertex)* if its removal disconnects the graph.

Definition 7.3 A *block* of a graph $\mathcal{G}$ is a maximal connected subgraph of $\mathcal{G}$ that has no cut-vertex.

Remark 21 Two blocks in a graph share at most one vertex. Hence the blocks of a graph partition its edge set. Furthermore, a vertex shared by two blocks must be a cut-vertex.

Theorem 7.5 *Every pure-strategy Nash equilibrium of a 2-player diffusion game must lie within one of the blocks of its underlying network.*

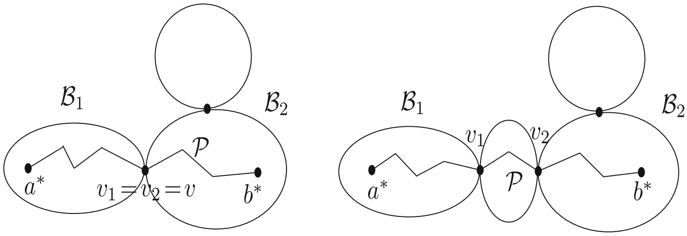


Fig. 7.5 Illustration of block diagrams in Theorem 7.5

Proof Given a network $\mathcal{G} = (V, \mathcal{E})$ with block decomposition $\mathcal{B}_1, \mathcal{B}_2, \ldots, \mathcal{B}_k$, let us denote one of the pure-strategy Nash equilibria of the diffusion game on $\mathcal{G}$ by $(a^*, b^*) \in V \times V$. If $a^*, b^* \in \mathcal{B}_i$, for some $i \in [k]$, then there is nothing to prove. Otherwise, without any loss of generality, let us assume that $a^* \in \mathcal{B}_1$ and $b^* \in \mathcal{B}_2$. Starting from node a^* and moving along a shortest path $\mathcal{P}$ between a^* and b^*, the path must exit block $\mathcal{B}_1$ for the first time at some vertex v_1. Clearly by Remark 21 such a vertex must be a cut vertex since it is shared between two blocks. By a similar argument, but this time by starting from node b^* and moving along the path $\mathcal{P}$, we can see that the path $\mathcal{P}$ must exit block $\mathcal{B}_2$ through a cut vertex v_2. Next we consider two cases:

- $v_1 = v_2 := v$. In this case, we can assume that neither $a^* = v$ nor $b^* = v$; otherwise, (a^*, b^*) would already be an equilibrium within one block (either $\mathcal{B}_1$ or $\mathcal{B}_2$). Now, without any loss of generality let us assume that the vertex v does not adopt type A for the seed placement (a^*, b^*). Then, the first player can strictly increase its utility by removing its seed from a^* and placing it in node v. In fact, placing the initial seeds on (v, b^*) instead of (a^*, b^*), the first player not only can adopt all the nodes that he was able to adopt in (a^*, b^*) (this is due to the fact that there is no edge between vertices of two distinct blocks (Remark 21)), but also he can adopt at least one more new node, which is node v. This is in contrast with (a^*, b^*) being a Nash equilibrium. Hence (a^*, b^*) must lie within one block.
- $v_1 \neq v_2$. In this case, by the definition of the cut vertices v_1 and v_2 we clearly have $d_{\mathcal{G}}(a^*, v_1) < d_{\mathcal{G}}(a^*, v_2)$, and $d_{\mathcal{G}}(b^*, v_2) < d_{\mathcal{G}}(b^*, v_1)$ (Fig. 7.5). Now, if v_2 adopts type A, by Lemma 7.1 it is straightforward to see that v_1 adopts type A as well. But then by the same argument as in the first case, the second player can move its seed from b^* to v_1 and strictly increase its utility. On the other hand, if v_2 does not adopt type A, then the first player can move its seed from a^* to v_2, and strictly increase its utility. This is in contradiction with (a^*, b^*) being a Nash equilibrium, and hence (a^*, b^*) must lie within one block. □

We note that Theorem 7.5 breaks down the complication of search for a Nash equilibrium over the entire graph to only its blocks. We further note that there are efficient time algorithms which can decompose a graph into its blocks in at most $O(n^2)$ steps, where n is the number of the nodes in the graph [10].

7.6 The Diffusion Game over Random Graphs

Social networks that are observed in the real world can be viewed as a single realization of an underlying stochastic process. This line of thinking has generated a huge interest in modeling real world social networks using random networks. In fact, the Erdos-Renyi graph $G(n, p)$, where there are n nodes and the edges emerge independently with probability $p \in (0, 1)$, is one of the standard models of random graphs as an instance of small world networks, in which most of the social entities can be reached from every other by a small number of hops or steps. Here, it is worth noting that although Erdos-Renyi graphs do not reflect some of the features inherited in real world social networks, such as having a high number of triangles, still, their simple definition makes them a popular model in studying social and random networks. In particular, analyzing such networks is beneficial in the sense that often the analytic results derived for Erdos-Renyi graphs can be leveraged for analysis of more complex random structures.

In this section we consider a two player diffusion game with single seed placement over the Erdos-Renyi graph $G(n, p)$. It is a well-known fact [11, Theorem 1] that $p(n) = \frac{\ln n}{n}$ is a threshold function for the connectivity of the random graph $G(n, p)$, meaning that as $n \to \infty$, the probability of the event that $G(n, \frac{c \ln n}{n})$ is connected tends to 1, for any fixed constant $c > 1$, and tends to 0, for any $c < 1$ (the probability $p(n)$ used here refers to $N(n) = \binom{n}{2} p(n)$ in [11]). In particular, for $p(n) \geq \frac{c \ln n}{n}$, $c > 1$, almost surely there exists no isolated vertex in $G(n, p(n))$, as $n \to \infty$. On the other hand, it was shown earlier that $p(n) = \sqrt{\frac{2 \ln n}{n}}$ is a threshold function for having diameter 2 in $G(n, p)$. In particular, for any fixed constant $c > 1$ almost all the nodes in $G(n, \sqrt{\frac{(1+c) \ln n}{n}})$ lie within a graphical distance of at most 2 from each other, which results in some straightforward analysis of the diffusion game over such graphs. Therefore, in this section we confine our attention to the more interesting region where $p \in (\frac{c \ln n}{n}, \sqrt{\frac{(1+c) \ln n}{n}})$, $c > 1$.

For any arbitrary but fixed node x and any realization $\mathcal{G}$ of $G(n, p)$, we let $S_{\mathcal{G}}(i)$ and $B_{\mathcal{G}}(i)$ be the sets of all the nodes which are, respectively, at graphical distances of exactly i, and at most i from node x. Similarly, we define $S(i)$ and $B(i)$ be two random sets denoting, respectively, the set of nodes of distances exactly i, and at most i from node x when the underlying graph is a random graph $G(n, p)$. Now we have the following lemma.

Lemma 7.3 *For an arbitrary $\lambda > (n-1)p$, and any $i \in [n]$ we have*

$$\mathbb{P}(|S(i)| \geq \lambda | B(i-1)|) \leq n e^{-\frac{(\lambda-(n-1)p)^2}{3(n-1)p}}.$$

Proof

$$\begin{aligned}\mathbb{P}\Big(|S(i)| \geq \lambda|B(i-1)|\Big) &= \mathbb{P}\left(\frac{|S(i)|}{|B(i-1)|} \geq \lambda\right)\\ &\leq \mathbb{P}\Big(\exists v \in B(i-1) : d(v) \geq \lambda\Big) \leq \mathbb{P}\Big(\exists v : d(v) \geq \lambda\Big)\\ &\leq n\mathbb{P}(d(v) \geq \lambda) = n\mathbb{P}\Big(d(u) - (n-1)p \geq \lambda - (n-1)p\Big)\\ &\leq ne^{-\frac{(\lambda-(n-1)p)^2}{3(n-1)p}},\end{aligned}$$

where in the first inequality we have used the fact that every node in $S(i)$ must be connected to at least one vertex in $B(i-1)$. Therefore, there exists at least one vertex in $B(i-1)$ whose degree is greater than or equal to $\frac{|S(i)|}{|B(i-1)|}$. Finally, in the last inequality we have used the Chernoff bound for the random variable $d(v)$. Note that $d(v)$ is the sum of $(n-1)$ independent Bernoulli random variables with equal probability of occurrence p.

Now we are ready to state the main result of this section.

Theorem 7.6 *For any arbitrary constants $\alpha, c > 1$, let $p \in [\frac{c \ln n}{n}, \sqrt{\frac{(1+c)\ln n}{n^{\alpha}}}]$. Then, as $n \to \infty$, for every random seed placement in the 2-player diffusion game over $G(n, p)$ with single seed, we have $\mathbb{E}[U_A] = \mathbb{E}[U_B] \geq \frac{1}{5p}$, and $\frac{U_A}{\mathbb{E}[U_A]} = \frac{U_B}{\mathbb{E}[U_B]} \to 1$.*

Proof For an arbitrary but fixed node x, let $I(x)$ be the event that in the random graph $G(n, p)$ with uniform seed placements at $(a, b) \in V \times V$, node x adopts either of the two types, and we denote its complement by $I^c(x)$. Conditioning on $G(n, p) = \mathcal{G}$, we can write

$$\begin{aligned}\mathbb{P}(I(x)|G(n,p) = \mathcal{G}) &\geq \frac{\sum_{i \neq j} |S_{\mathcal{G}}(i)||S_{\mathcal{G}}(j)|}{n^2}\\ \mathbb{P}(I^c(x)|G(n,p) = \mathcal{G}) &\leq \frac{\sum_{i=1}^{n} |S_{\mathcal{G}}(i)|^2}{n^2},\end{aligned} \tag{7.4}$$

where the first inequality is due to the fact that if the seed nodes lie in different sets $S_{\mathcal{G}}(i), S_{\mathcal{G}}(i), i \neq j$ from node x, then by Lemma 7.1 node x will adopt one of the types. Moreover, for the second inequality and using Lemma 7.1, one can see that the set of uninfected or gray nodes is a subset of vertices of equal distance from the seed nodes.

Now let us define the following sets of graphs:

$$\mathcal{K} = \{\mathcal{G} : d_{\mathcal{G}}(v) > 0, \forall v\}, \qquad \mathcal{F} = \{\mathcal{G} : |S_{\mathcal{G}}(i)| < \lambda|B_{\mathcal{G}}(i-1)|, \forall i \in [n]\}.$$

By combining the two inequalities in (7.4) and taking the average over the probability space of all possible graphs with distribution $G(n, p)$, we can write

$$\frac{\mathbb{P}(I^c(x))}{\mathbb{P}(I(x))} = \frac{\sum_{\mathcal{G}} \mathbb{P}(\mathcal{G})\mathbb{P}(I^c(x)|G(n,p)=\mathcal{G})}{\sum_{\mathcal{G}} \mathbb{P}(\mathcal{G})\mathbb{P}(I(x)|G(n,p)=\mathcal{G})} \leq \frac{\sum_{\mathcal{G}} \mathbb{P}(\mathcal{G}) \left(\sum_{i=1}^{n} |S_{\mathcal{G}}(i)|^2\right)}{\sum_{\mathcal{G}} \mathbb{P}(\mathcal{G}) \left(\sum_{i\neq j} |S_{\mathcal{G}}(i)||S_{\mathcal{G}}(j)|\right)}$$

$$= \frac{\sum\limits_{\mathcal{G}\in\mathcal{F}} \mathbb{P}(\mathcal{G})\left(\sum\limits_{i=1}^{n} |S_{\mathcal{G}}(i)|^2\right) + \sum\limits_{\mathcal{G}\in\mathcal{F}^c} \mathbb{P}(\mathcal{G})\left(\sum\limits_{i=1}^{n} |S_{\mathcal{G}}(i)|^2\right)}{\sum\limits_{\mathcal{G}} \mathbb{P}(\mathcal{G})\left(\sum\limits_{i\neq j} |S_{\mathcal{G}}(i)||S_{\mathcal{G}}(j)|\right)}$$

$$\leq \frac{\sum\limits_{\mathcal{G}\in\mathcal{F}} \mathbb{P}(\mathcal{G})\left(\sum\limits_{i=1}^{n} |S_{\mathcal{G}}(i)|^2\right) + n^2\mathbb{P}\left(\mathcal{G}\in\mathcal{F}^c\right)}{\sum\limits_{\mathcal{G}\in\mathcal{F}} \mathbb{P}(\mathcal{G})\left(\sum\limits_{i\neq j} |S_{\mathcal{G}}(i)||S_{\mathcal{G}}(j)|\right)}, \tag{7.5}$$

where the second inequality holds because $\sum_{i=1}^{n} |S_{\mathcal{G}}(i)|^2 \leq n^2$. On the other hand, we have

$$\sum_{i\neq j} |S_{\mathcal{G}}(i)||S_{\mathcal{G}}(j)| = 2\sum_{i=1}^{n} |S_{\mathcal{G}}(i)| \sum_{j\leq i-1} |S_{\mathcal{G}}(j)| = 2\sum_{i=1}^{n} |S_{\mathcal{G}}(i)||B_{\mathcal{G}}(i-1)|. \tag{7.6}$$

Using (7.6) in (7.5) we can write

$$\frac{\mathbb{P}(I^c(x))}{\mathbb{P}(I(x))} \leq \frac{\sum\limits_{\mathcal{G}\in\mathcal{F}} \mathbb{P}(\mathcal{G})\left(\sum\limits_{i=1}^{n} |S_{\mathcal{G}}(i)|^2\right) + n^2\mathbb{P}\left(\mathcal{G}\in\mathcal{F}^c\right)}{2\sum\limits_{\mathcal{G}\in\mathcal{F}} \mathbb{P}(\mathcal{G})\sum\limits_{i=1}^{n} |S_{\mathcal{G}}(i)||B_{\mathcal{G}}(i-1)|}$$

$$\leq \frac{\sum\limits_{\mathcal{G}\in\mathcal{F}} \mathbb{P}(\mathcal{G})\left(\sum\limits_{i=1}^{n} |S_{\mathcal{G}}(i)|^2\right) + n^2\mathbb{P}\left(\mathcal{G}\in\mathcal{F}^c\right)}{\frac{2}{\lambda}\sum\limits_{\mathcal{G}\in\mathcal{F}} \mathbb{P}(\mathcal{G})\left(\sum\limits_{i=1}^{n} |S_{\mathcal{G}}(i)|^2\right)}$$

$$= \frac{\lambda}{2} + \frac{n^2\lambda}{2}\frac{\mathbb{P}\left(\mathcal{G}\in\mathcal{F}^c\right)}{\sum\limits_{\mathcal{G}\in\mathcal{F}} \mathbb{P}(\mathcal{G})\left(\sum\limits_{i=1}^{n} |S_{\mathcal{G}}(i)|^2\right)}$$

$$\leq \frac{\lambda}{2} + \frac{n^2\lambda}{2}\frac{\mathbb{P}\left(\mathcal{G}\in\mathcal{F}^c\right)}{\sum\limits_{\mathcal{G}\in\mathcal{F}\cap\mathcal{K}} \mathbb{P}(\mathcal{G})\left(\sum\limits_{i=1}^{n} |S_{\mathcal{G}}(i)|^2\right)}$$

$$\leq \frac{\lambda}{2} + \frac{n^2\lambda}{2}\frac{\mathbb{P}\left(\mathcal{G}\in\mathcal{F}^c\right)}{\mathbb{P}\left(\mathcal{G}\in\mathcal{F}\cap\mathcal{K}\right)}, \tag{7.7}$$

where in the second inequality we have used the definition of set $\mathcal{F}$, and in the last inequality we have used the definition of $\mathcal{K}$ to get $\sum_{i=1}^{n} |S_{\mathcal{G}}(i)|^2 \geq 1, \forall \mathcal{G} \in \mathcal{F} \cap \mathcal{K}$. On the other hand, by Lemma 7.3 and using the union bound, we can write

$$\begin{aligned}\mathbb{P}(\mathcal{G} \in \mathcal{F}^c) &= \mathbb{P}(\exists i \in [n] : |S(i)| \geq \lambda |B(i-1)|)\\ &\leq \sum_{i}^{n} \mathbb{P}(|S(i)| \geq \lambda |B(i-1)|) \leq n^2 e^{-\frac{(\lambda-(n-1)p)^2}{3(n-1)p}}. \end{aligned} \tag{7.8}$$

By choosing $\lambda = (1+\sqrt{15})np$ for sufficiently large n, and using (7.8), each of the probabilities $\mathbb{P}(\mathcal{G} \in \mathcal{F})$ and $\mathbb{P}(\mathcal{G} \in \mathcal{K})$ can be made arbitrarily close to 1. Therefore, for sufficiently large n, we have $\mathbb{P}(\mathcal{G} \in \mathcal{F} \cap \mathcal{K}) \geq \frac{1}{2}$. Hence, substituting (7.8) in (7.7) and using $\mathbb{P}(\mathcal{G} \in \mathcal{F} \cap \mathcal{K}) \geq \frac{1}{2}$, we get

$$\begin{aligned}\frac{\mathbb{P}(I^c(x))}{\mathbb{P}(I(x))} &\leq \frac{\lambda}{2} + \lambda n^4 e^{-\frac{(\lambda-(n-1)p)^2}{3(n-1)p}} = \frac{1+\sqrt{15}}{2} np + (1+\sqrt{15}) n^5 p e^{-\frac{((1+\sqrt{15})np-(n-1)p)^2}{3(n-1)p}}\\ &\leq \frac{1+\sqrt{15}}{2} np + (1+\sqrt{15}) n^5 p e^{-\frac{15 \ln n}{3}} = \frac{1+\sqrt{15}}{2} np + (1+\sqrt{15}) p,\end{aligned}$$

where the second inequality holds because $p \geq \frac{\ln n}{n}$. Finally, since $\mathbb{P}(I^c(x)) = 1 - \mathbb{P}(I(x))$, we get $\mathbb{P}(I(x)) \geq \frac{2}{2+(2+2\sqrt{15})p+(1+\sqrt{15})np}$. Now due to the symmetry between players we have

$$\mathbb{E}[U_A] = \mathbb{E}[U_B] = \frac{1}{2}\mathbb{E}[U_A + U_B] = \frac{1}{2}\sum_{x} P(I(x)) \geq \frac{n}{2+(2+2\sqrt{15})p+(1+\sqrt{15})np}.$$

Therefore, for sufficiently large n, we have $\mathbb{E}[U_A] \geq \frac{1}{5p}$.

Now for two arbitrary but fixed nodes a and b, let us assume that players A and B place their seeds at a and b, respectively. We introduce a martingale by setting $X_0 = \mathbb{E}[U_A(a,b)]$, and $X_i = \mathbb{E}[U_A(a,b) | v_1 = a, v_2 = b, v_3, v_4, \ldots, v_i]$ to be the expected utility of player A after the first i nodes of the random graph $G(n, p)$ are exposed. In other words, to find X_i we expose the first i vertices and all their internal edges and take the conditional expectation of U_A with that partial information. It is straightforward to check that this defines a martingale of length at most n such that $|X_{i+1} - X_i| \leq 1$, as adding one vertex can at most change the utility of player A by at most 1. Therefore, using Azumas' inequality (Lemma 2.10) we can write $\mathbb{P}\Big(|U_A - \mathbb{E}[U_A]| \geq \sqrt{n}\theta\Big) \leq 2e^{-\frac{\theta^2}{2}}$. In particular, since $\mathbb{E}[U_A] \geq \frac{1}{5p}$, we have

$$\mathbb{P}\Big(\Big|\frac{U_A}{\mathbb{E}[U_A]} - 1\Big| \geq 5p\sqrt{n}\theta\Big) \leq \mathbb{P}\Big(\Big|\frac{U_A}{\mathbb{E}[U_A]} - 1\Big| \geq \frac{\sqrt{n}\theta}{\mathbb{E}[U_A]}\Big) \leq 2e^{-\frac{\theta^2}{2}}.$$

By choosing $\theta = \ln n$ and since $p \in [\frac{c \ln n}{n}, \sqrt{\frac{(1+c)\ln n}{n^{1+\alpha}}}]$, $c > 1$, one can see that for any $\epsilon > 0$, there exists a sufficiently large $n(\epsilon)$ such that for $n \geq n(\epsilon)$, we have $\mathbb{P}\left(\left|\frac{U_A}{\mathbb{E}[U_A]} - 1\right| \geq \epsilon\right) \leq \epsilon$. By the same argument and by symmetry, we can see that for sufficiently large n, U_B is arbitrarily close to its mean. This completes the proof. □

As we close this section, we stress the fact that the subset of nodes which adopt either of the two types in diffusion games can be viewed as a community whose members have closer interactions with each other. In other words, in the diffusion game each community can be viewed as the final subset of nodes that adopt a specific technology, which raises the question of efficient decomposition of the network into different communities. In such problems, the main issue is to partition the set of nodes into different groups such that the set of edges within each group is much larger than that between the groups. Different approaches in order to determine the communities effectively so that they scale with the parameters of the network under both deterministic and randomized settings have been proposed in the literature [12, 13]. As an example, an electrical voltage-based approach in order to determine the communities within a network such that they scale linearly with the size of the network has been discussed in [14].

7.7 Social Welfare and Sub-modularity in the Competitive Diffusion Game

In this section we first study the maximum social welfare of the game which can be achieved by two players in the competitive diffusion game with a single seed placement. The motivation for such a study is that any bound which is obtained for the optimal social welfare can be used to provide some bound on the price of anarchy of the game, which measures the degradation in the efficiency of a system due to selfish behavior of its agents, and is defined to be the ratio of the centralized optimal social welfare over the sum utilities of the worst equilibrium. Following that, we study the sub-modular property in the competitive diffusion game. It was shown in the literature that greedy algorithm optimization approaches work quite well for the dynamics which benefit from sub-modular property [15]. In this section we show that unlike some other diffusion processes [16], the utilities of the players in the competitive diffusion game do not have the sub-modular property.

We start with the following definition:

Definition 7.4 Given an arbitrary nonnegative matrix P, we define its zero pattern $\sigma(P)$ to be

$$\sigma(P)_{ij} = \begin{cases} 1, & P_{ij} > 0 \\ 0, & P_{ij} = 0. \end{cases}$$

Theorem 7.7 *Given a graph $\mathcal{G} = (V, \mathcal{E})$ of n nodes and diameter D, and two players A and B, there exists an initial seed placement (a, b) for players such that the social utility $U_A(a, b) + U_B(a, b)$ is at least*

$$n + 1 - \frac{\sum_{k=1}^{D} \left\| \left(\sigma(\mathcal{A}^k) - \sigma(\mathcal{A}^{k-1})\right)\mathbf{1} \right\|^2}{n(n-1)},$$

where $\| \cdot \|$ denotes the standard Euclidean norm, and $\mathcal{A} = I + \mathcal{A}_{\mathcal{G}}$ where $\mathcal{A}_{\mathcal{G}}$ is the adjacency matrix of the network $\mathcal{G}$.

Proof Let us define $\mathcal{G}^{(k)} = (V, \mathcal{E}^{(k)})$, where $\mathcal{E}^{(k)} = \{(i, j) | d_{\mathcal{G}}(i, j) = k\}$. We consider all the initial placements over different pairs of nodes, and then compute the average utility gained by players. To do that, we consider an array Q of $\binom{n}{2}$ rows and n different columns. For $i \neq j, k \in \{1, 2, \ldots n\}$, we let $Q(\{i, j\}, k) = 1$ if and only if node k adopts either A or B during the diffusion process for the initial placement $\{i, j\}$, and $Q(\{i, j\}, k) = 0$, otherwise (Fig. 7.6). We count the maximum number of zeros in Q. For an arbitrary but fixed node $x \in V = \{1, 2, \ldots, n\}$, we count the number of different initial seed placements which result in node x turning to gray. In other words, we count the maximum number of zeros in column x of Q. For this purpose we note that, using Lemma 7.1, if node x turns to gray, it must be equidistant from seed nodes. On the other hand, for a given $k = 1, 2, \ldots, D$, the number of choosing two nodes at distance k from node x (as possible seed placements which may turn x to gray) is the same as the number of choosing two nodes among neighbors of x in $\mathcal{G}^{(k)}$, i.e. $\binom{d_{\mathcal{G}^{(k)}}(x)}{2}$, where $d_{\mathcal{G}^{(k)}}(x)$ denotes the degree of node x in $\mathcal{G}^{(k)}$. Thus, the maximum number of zeros in column x of Q is upper bounded by $\sum_{k=1}^{D} \binom{d_{\mathcal{G}^{(k)}}(x)}{2}$ and hence the number of zeros in Q is bounded from above by $\sum_{x \in V} \sum_{k=1}^{D} \binom{d_{\mathcal{G}^{(k)}}(x)}{2}$. This means that the average number of ones in each row of Q is at least

Fig. 7.6 Illustration of the array Q in Theorem 7.7

	1	2	3	4	⋯		n
(1, 2)	1	1	0	1		1	1
(1, 3)	1	0	1	0		1	0
(1, 4)	1	0	0	1		0	0
(2, 3)	0	1	1	0		0	1
(2, 4)	0	1	0	1		1	1
(3, 4)	0	0	1	1		0	0
⋮					⋱		
$(n-1, n)$	0	1	1	0		1	1

$$\begin{aligned}\frac{n\binom{n}{2}-\sum_{x\in V}\sum_{k=1}^{D}\binom{d_{\mathcal{G}^{(k)}}(x)}{2}}{\binom{n}{2}} &= n-\frac{\sum_{k=1}^{D}\sum_{x\in V}\binom{d_{\mathcal{G}^{(k)}}(x)}{2}}{\binom{n}{2}}\\ &= n+\frac{\sum_{k=1}^{D}\sum_{x\in V}d_{\mathcal{G}^{(k)}}(x)}{n(n-1)}-\frac{\sum_{k=1}^{D}\sum_{x\in V}d^2_{\mathcal{G}^{(k)}}(x)}{n(n-1)}\\ &= n+\frac{\sum_{k=1}^{D}2|\mathcal{E}^{(k)}|}{n(n-1)}-\frac{\sum_{k=1}^{D}\sum_{x\in V}d^2_{\mathcal{G}^{(k)}}(x)}{n(n-1)}\\ &= n+1-\frac{\sum_{k=1}^{D}\sum_{x\in V}d^2_{\mathcal{G}^{(k)}}(x)}{n(n-1)},\end{aligned}\tag{7.9}$$

where the last equality follows because $\{\mathcal{E}^{(k)}\}_{k=1}^{D}$ partitions all the edges of a complete graph with n nodes. This yields a lower bound on the maximum social welfare for the case of two players on the graph.

Finally, using the zero pattern definition of a matrix, we can compute the last quantity in (7.9) in the following way. Let us take $A_{\mathcal{G}}$ to be the adjacency matrix of graph $\mathcal{G}$ of diameter D and $\mathcal{A}=I+\mathcal{A}_{\mathcal{G}}$, where I denotes the identity matrix of appropriate dimension. It is not hard to see that $\sigma(\mathcal{A}^k)-\sigma(\mathcal{A}^{k-1})$ is the adjacency matrix of $\mathcal{G}^{(k)}$. In other words, $\mathcal{A}_{\mathcal{G}^{(k)}}=\sigma(\mathcal{A}^k)-\sigma(\mathcal{A}^{k-1})$. Therefore, if we let $\mathbf{1}$ be the column vector of all ones, the degree of each node x in $\mathcal{G}^{(k)}$ can be found easily by looking at the x coordinate of vector $\big[\sigma(\mathcal{A}^k)-\sigma(\mathcal{A}^{k-1})\big]\mathbf{1}$. Thus we can write

$$\begin{aligned}\sum_{x\in V}\sum_{k=1}^{D}d^2_{\mathcal{G}^{(k)}}(x) &= \sum_{k=1}^{D}\mathbf{1}'\big[\sigma(\mathcal{A}^k)-\sigma(\mathcal{A}^{k-1})\big]'\big[\sigma(\mathcal{A}^k)-\sigma(\mathcal{A}^{k-1})\big]\mathbf{1}\\ &= \sum_{k=1}^{D}\left\|\,\big(\sigma(\mathcal{A}^k)-\sigma(\mathcal{A}^{k-1})\big)\,\mathbf{1}\right\|^2.\end{aligned}$$

□

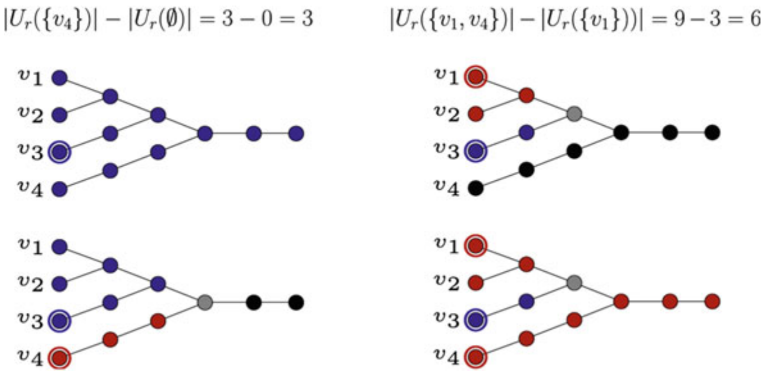

$|U_r(\{v_4\})|-|U_r(\emptyset)|=3-0=3$
$|U_r(\{v_1,v_4\})|-|U_r(\{v_1\})|=9-3=6$
v_1
v_2
v_3
v_4
v_1
v_2
v_3
v_4
v_1
v_2
v_3
v_4
v_1
v_2
v_3
v_4

Fig. 7.7 A counterexample for the existence of sub-modular property

Finally, the following example which is due to Esther Galbrun shows that the sub-modular property does not hold for the diffusion game.

Example 7.1 (*A Counterexample by Esther Galbrun*) In this example, there are two players red (r) and blue (b). Circled nodes are initial seeds and the graph shows the color of the nodes at the end of the diffusion. In all cases blue only picks v_3. Moreover, $S = \emptyset \subseteq \{v_1\} = \bar{S}$ and $x = \{v_4\}$. As can be seen in Fig. 7.7, the utility function of the red player does not satisfy sub-modular property.

7.8 Conclusion

In this chapter, we have studied a class of games known as diffusion games which model the competitive behavior of a set of social actors on an undirected connected social network. We determined the set of pure-strategy Nash equilibria for two special but well-studied classes of networks. We showed that, in general, making a decision on the existence of Nash equilibrium for such a class of games is an NP-hard problem. Further, we have presented some necessary conditions for a given profile to be an equilibrium in general graphs. We have studied the behavior of the competitive diffusion game over Erdos-Renyi graphs, obtained some concentration results, and derived lower bounds for the expected utilities of the players over such random structures. Finally, we have provided a lower bound for the social welfare in such games and have shown that the utility of the players are in general not sub-modular.

References

1. N. Alon, M. Feldman, A. D. Procaccia, and M. Tennenholtz, "A note on competitive diffusion through social networks," *Information Processing Letters*, vol. 110, no. 6, pp. 221–225, 2010.
2. S. R. Etesami and T. Başar, "Complexity of equilibrium in competitive diffusion games on social networks," *Automatica*, vol. 68, pp. 100–110, 2016.
3. S. R. Etesami and T. Başar, "Complexity of equilibrium in diffusion games on social networks," *Proc. 2014 American Control Conference (ACC 2014), Portland, OR, June 4–6, 2014*, pp. 2077–2082, 2014.
4. M. Fazli, M. Ghodsi, J. Habibi, P. J. Khalilabadi, V. Mirrokni, and S. S. Sadeghabad, "On the non-progressive spread of influence through social networks," in *LATIN 2012: Theoretical Informatics*. Springer, 2012, pp. 315–326.
5. L. Small and O. Mason, "Information diffusion on the iterated local transitivity model of online social networks," *Discrete Applied Mathematics*, 2012.
6. R. Takehara, M. Hachimori, and M. Shigeno, "A comment on pure-strategy Nash equilibria in competitive diffusion games," *Information Processing Letters*, vol. 112, no. 3, pp. 59–60, 2012.
7. L. Small and O. Mason, "Nash equilibria for competitive information diffusion on trees," *Information Processing Letters*, vol. 113, no. 7, pp. 217–219, 2013.
8. C. Dürr and N. K. Thang, "Nash equilibria in Voronoi games on graphs," in *Algorithms-ESA 2007*. Springer, 2007, pp. 17–28.

9. M. R. Garey and D. S. Johnson, “Complexity results for multiprocessor scheduling under resource constraints,” *SIAM Journal on Computing*, vol. 4, no. 4, pp. 397–411, 1975.
10. K. Paton, “An algorithm for the blocks and cutnodes of a graph,” *Communications of the ACM*, vol. 14, no. 7, pp. 468–475, 1971.
11. P. Erdos and A. Renyi, “On random graphs I,” in *Publ. Math. Debrecen*, vol. 6, 1959, pp. 290–297.
12. B. Hajek, Y. Wu, and J. Xu, “Computational lower bounds for community detection on random graphs,” *arXiv preprint* arXiv:1406.6625, 2014.
13. M. E. J. Newman, “Detecting community structure in networks,” *Eur. Phys. J. B*, vol. 38, pp. 321–330, 2004.
14. F. Wu and B. A. Huberman, “Finding communities in linear time: A physics approach,” in http://arxiv.org/abs/cond-mat/0310600, 2003.
15. E. Mossel and S. Roch, “Submodularity of influence in social networks: From local to global,” *SIAM Journal on Computing*, vol. 39, no. 6, pp. 2176–2188, 2010.
16. 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*. ACM, 2003, pp. 137–146.

Chapter 8
Conclusions and Directions for Future Research

8.1 Conclusions

In this thesis, we have discussed four different types of popular social, communication and distributed networks. We have grouped the problems into two main categories: distributed control problems where the agents interact locally in order to achieve a certain goal, and game problems where the agents act in a strategic manner in order to maximize their own utilities. In fact, we have studied a set of problems which lie at the intersection of these two categories. We have discussed existing relevant results from the literature, and presented several new results. In particular, we have noted how a well-constructed function serving as a potential function can facilitate analysis of such networks. Using such functions, we were able to evaluate the convergence speed of the dynamics toward an ultimate goal. Furthermore, we have leveraged such functions in order to devise more effective and faster algorithms which drive the entire network from an initial state to a final outcome.

8.2 Directions for Future Research

Each of the problems considered in this thesis opens an avenue for future directions of research as we will discuss in the following:

- Related to the unbiased quantized consensus problem posed in Chap. 3, an interesting problem is to study the expected convergence time of such dynamics in the absence of reversibility of the transition probabilities. In fact, our analysis in that chapter heavily relied on the reversibility of the transition probabilities of the underlying dynamics. Therefore, obtaining similar results for non-reversible chains will be an interesting challenge. Furthermore, as a network designer, studying the convergence properties and the expected convergence time of such dynamics in the presence of an adversary would constitute an important problem. As an example,

S.R. Etesami, *Potential-Based Analysis of Social, Communication, and Distributed Networks*, Springer Theses, DOI 10.1007/978-3-319-54289-8_8

given a network $\mathcal{G}$, one can think of adding an edge (or removing an edge) so as to minimize (or maximize) the expected convergence time.

- Related to the Metropolis quantized consensus introduced in Chap. 4, one can consider to improve convergence times even further by modifying the underlying dynamics. For example, [1] attained a linear convergence time for consensus on any fixed graph, and it is an open question to obtain a quantized consensus protocol which replicates this. A related problem is to extend the quadratic convergence times obtained in this chapter to dynamic networks which are not necessarily connected at every time, but rather only connected in a long-term sense.
- Inspired by the results given in Sect. 5.3, we would like to know whether for the case of heterogeneous Hegselmann-Krause dynamics there is any way to design a proper utility function for each player such that the resulting network formation game converts to a team problem, such that each player's update contributes an increase (decrease) to a global function toward an equilibrium. This is a widely open problem which has numerous applications on understanding the complex nature of opinion formation in biased societies (societies with nonidentical entities). Moreover, as we have seen in Chap. 5, the dynamics of the Hegselmann-Krause model are not robust with respect to changes. In other words, any small change in the initial opinion of the agents can result in substantially different outcomes. Therefore, one may think of how to enrich the Hegselmann-Krause model in order to remove some of its current limitations.
- Related to the capacitated selfish replication (CSR) game posed in Chap. 6, one can determine the complexity of search for a pure-strategy Nash equilibrium. Moreover, in order to find an equilibrium in the binary CSR game, one can increase the cache sizes and recover an equilibrium very fast using the least best response dynamics. Therefore, one natural question is the relationship of this equilibrium with that of the unit size cache CSR game. For example, one may be able to construct an equilibrium for the unit size cache CSR game using an equilibrium of a CSR game with slightly higher capacities and use some majority rules in order to assign only one unit resource to each player. Also one can consider extensions of the results given in this chapter to more dynamic settings when the communication network itself is subject to a change, see, e.g., [2, 3].
- Related to the diffusion game model introduced in Chap. 7, an interesting problem is identifying the class of networks which admit pure-strategy Nash equilibria for the case of two players in the competitive diffusion game. It is not hard to see that a tree construction which is a special case of bipartite graphs always leads to a pure-strategy equilibrium [4] for the case of two players. Moreover, we would like to study the continuous version of such games where the players are allowed to randomize among their possible actions, as it can relax some of the existing hardness limitations in the current form of the diffusion game. Finally, studying a more robust model of the diffusion game with respect to changes in the network topology is another important problem to pursue.

References

1. A. Olshevsky, “Linear time average consensus on fixed graphs and implications for decentralized optimization and multi-agent control,” 2014. [Online]. Available: http://arxiv.org/abs/1411.4186.
2. A. Fabrikant, A. Luthra, E. Maneva, C. H. Papadimitriou, and S. Shenker, “On a network creation game,” in *Proceedings of the Twenty-Second Annual Symposium on Principles of Distributed Computing*. ACM, 2003, pp. 347–351.
3. S. R. Etesami and T. Başar, “Game-theoretic analysis of the Hegselmann-Krause model for opinion dynamics in finite dimensions,” *IEEE Transactions on Automatic Control, 60(7):1886-1897, July 2015*.
4. L. Small and O. Mason, “Nash equilibria for competitive information diffusion on trees,” *Information Processing Letters*, vol. 113, no. 7, pp. 217–219, 2013.

Appendix A
Complementary Results for Chapter 3

In this Appendix, we provide some preliminary and complementary results for the material given in Chap. 3.

Lemma A.1 *For the star graph and the double-star graph with m edges, the maximum expected meeting time $\overline{T}$ is bounded from above by $O(m^2)$.*

Proof Let us denote the meeting time function of the original process by $M^{\mathrm{o}}(\cdot,\cdot) : V \times V \to \mathbb{R}$. For the case of star graph with m edges, a center node z, and two leaves x and y, it is not hard to see that $\overline{T} = M^{\mathrm{o}}(x, y)$. Because of the symmetric structure of the star graph and by one-step recursive expansion of the meeting time function, we can write

$$\begin{aligned} M^{\mathrm{o}}(x, y) &= \frac{2}{m}(1 + M^{\mathrm{o}}(x, z)) + \frac{m-2}{m}(1 + M^{\mathrm{o}}(x, y)) \\ M^{\mathrm{o}}(x, z) &= \frac{1}{m} + \frac{m-1}{m}(1 + M^{\mathrm{o}}(x, y)). \end{aligned}$$

Solving these two equations we get $\overline{T} = M^{\mathrm{o}}(x, y) = \frac{m(m+2)}{2}$. For the case of the double-star graph, we use a similar line of argument. In a general form, we consider a double-star graph with center nodes x_1 and y_1 that share k neighbors, for some $k \geq 0$. Moreover, we assume that x_1 and y_1 have i and $m+1-i$ neighbors, respectively. Such a graph has been depicted in Fig. A.1. Again, using the symmetry, one can distinguish between 13 different states for the position of the walkers in such a graph. As an example, denoting the location of the two walkers by x, y and looking at Fig. A.1, one can observe that when $x \in N(x_1) \setminus N(y_1)$ the relative position of the other walker with respect to x can fit into one of the following cases:

$$y \in N(y_1) \setminus N(x_1), \quad y = y_1, \quad y \in N(x_1) \cap N(y_1), \quad y = x_1, \quad y \in N(x_1) \setminus N(y_1).$$

Note that in order to write recursion expansions of the expected meeting time of the original process, and due to symmetry, only the relative positions of the walkers matters. For example for $x \neq y$, $M^{\mathrm{o}}(x, y)$ is the same for all pairs of $x, y \in N(x_1)$.

S.R. Etesami, *Potential-Based Analysis of Social, Communication, and Distributed Networks*, Springer Theses, DOI 10.1007/978-3-319-54289-8

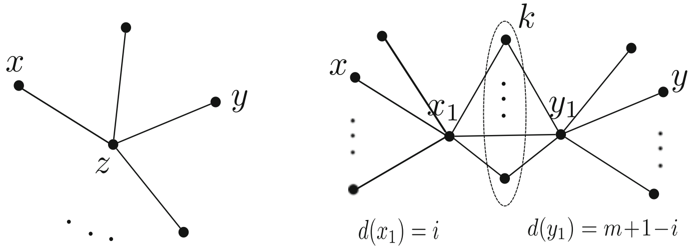


Fig. A.1 Star graph and double-star graph with *m* edges

Therefore, by recursion expansion of the expected meeting time of the original process for such a graph, we obtain a linear system of equations which for simplicity we write in a matrix form as has been shown in (A.1). Solving this system of equations fully characterizes the expected meeting time of the original process for being in different states of the double-star graph which are upper bounded by $O(\frac{i+m^2}{k})$. Since, $k \leq i \leq m$, for the double-star graph we get $\overline{T} = O(m^2)$.

$$\begin{pmatrix}
-\frac{2}{m} & \frac{1}{m} & \frac{2}{m} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-\frac{k-i+1}{m} & -\frac{i+1}{m} & 0 & 0 & \frac{1}{m} & 0 & 0 & \frac{k}{m} & 0 & 0 & 0 & 0 & \frac{1}{m} \\
-\frac{i+k-m}{m} & 0 & -\frac{m-i+2}{m} & 0 & 0 & 0 & \frac{k}{m} & 0 & \frac{1}{m} & 0 & 0 & 0 & \frac{1}{m} \\
0 & 0 & 0 & -\frac{m-i+2}{m} & 0 & \frac{1}{m} & 0 & -\frac{i+k-m}{m} & 0 & 0 & \frac{k-1}{m} & 0 & \frac{1}{m} \\
0 & \frac{1}{m} & 0 & 0 & -\frac{m-i+1}{m} & 0 & 0 & \frac{k}{m} & 0 & 0 & 0 & -\frac{i+k-m+1}{m} & 0 \\
0 & 0 & 0 & \frac{1}{m} & 0 & -\frac{i+1}{m} & -\frac{k-i+1}{m} & 0 & 0 & 0 & \frac{k-1}{m} & 0 & \frac{1}{m} \\
0 & 0 & \frac{1}{m} & 0 & 0 & \frac{1}{m} & -\frac{3}{m} & 0 & \frac{1}{m} & 0 & 0 & 0 & 0 \\
0 & \frac{1}{m} & 0 & \frac{1}{m} & \frac{1}{m} & 0 & 0 & -\frac{3}{m} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{1}{m} & 0 & 0 & 0 & \frac{k}{m} & 0 & -\frac{i}{m} & -\frac{k-i+2}{m} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{2}{m} & -\frac{2}{m} & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{2}{m} & 0 & \frac{2}{m} & 0 & 0 & 0 & 0 & -\frac{4}{m} & 0 & 0 \\
0 & 0 & 0 & 0 & \frac{2}{m} & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{2}{m} & 0 \\
0 & -\frac{i+k-m}{m} & -\frac{k-i+1}{m} & \frac{k}{m} & 0 & \frac{k}{m} & 0 & 0 & 0 & 0 & 0 & 0 & -1
\end{pmatrix}$$

$$\times \begin{pmatrix} M^{\text{o}}(x, y) \\ M^{\text{o}}(x_1, y) \\ M^{\text{o}}(x, y_1) \\ \\ \\ \vdots \\ \vdots \\ \\ \\ \\ \\ \\ M^{\text{o}}(x_1, y_1) \end{pmatrix} = - \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}. \tag{A.1}$$

□

Proof of Lemma 3.3 We use the upper bound of $\alpha_1(\mathcal{L}_\mathcal{G}) \leq \max\{d(u)+d(v)|(u, v) \in \mathcal{E}(\mathcal{G})\}$ given in [1]. Since we already assumed $d(u) + d(v) \leq m, \forall(u, v) \in \mathcal{E}(\mathcal{G})$, we consider two cases:

- $\max\{d(u)+d(v)|(u,v) \in \mathcal{E}(\mathcal{G})\} \leq m-1$. Then, we simply get $\alpha_1(\mathcal{L}_{\mathcal{G}}) \leq m-1 \leq m-\frac{1}{2}$
- $\max\{d(u)+d(v)|(u,v) \in \mathcal{E}(\mathcal{G})\} = m$. Then, there exists $(u^*,v^*) \in \mathcal{E}(\mathcal{G})$ such that $d(u^*)+d(v^*) = m$. In this case we used the upper bound of $\alpha_1(\mathcal{L}_{\mathcal{G}}) \leq 2+\sqrt{(m-2)(s-2)}$ given in [2], where $s = \max\{d(u)+d(v)|(u,v) \in \mathcal{E}(\mathcal{G}) \setminus (u^*,v^*)\}$. Since $d(u^*)+d(v^*) = \max\{d(u)+d(v)|(u,v) \in \mathcal{E}(\mathcal{G}) = m$, there exists exactly one edge in $\mathcal{G} \setminus \{u^*,v^*\}$ such that for any $(u,v) \in \mathcal{E}(\mathcal{G}) \setminus (u^*,v^*)$ we must have $d(u)+d(v) \leq m-1$. This shows that $s \leq m-1$, and hence,

$$\alpha_1(\mathcal{L}_{\mathcal{G}}) \leq 2+\sqrt{(m-2)(m-3)} = 2+\sqrt{(m-\frac{5}{2})^2-\frac{1}{4}} < m-\frac{1}{2}.$$

Therefore, in both cases we have $\alpha_1(\mathcal{L}_{\mathcal{G}}) \leq m-\frac{1}{2}$. □

In the remaining of this Appendix, we develop an alternative approach in order to study the maximum expected convergence time of unbiased quantized consensus over static graphs. In particular, we identify the precise order of the maximum expected convergence time for the case of simple static graphs such as line graph and cycle. Here we note that, although the result of this appendix works well when we benefit from inherent symmetry in the underlying graph $\mathcal{G}$, in general it does not lead to an explicit tight bound based on the parameters of the network.

Definition A.1 A *birth-and-death chain* of length $n+1$ has state space $\Omega = \{0,1,\ldots,n\}$ such that in one step the state can increase or decrease by at most 1.

Lemma A.2 *Assume that $\mathcal{G}$ is a connected graph with diameter D. Then, $\overline{T}$ is bounded from above by the maximum hitting time of a birth-and-death chain of length $D+1$ and positive transition probabilities greater than $\frac{1}{m}$.*

Proof We partition all the different states of the above original coupled random walks (Definition 3.1) into different classes. Here, we refer to each state as a possible pair of positions of the walkers in the network, and denote the set of all the states by $\mathcal{X}(\mathcal{G})$. For each state x we define $d_{\mathcal{G}}(x^{(0)},x^{(2)})$ to be the length of the shortest path between walker 0 and walker 2 in state x. Let

$$S_\ell = \{x \in \mathcal{X}(\mathcal{G})|\ d_{\mathcal{G}}(x^{(0)},x^{(2)}) = \ell\},\ \ell = 0,1,\ldots,D.$$

It is clear that $\{S_\ell\}_{\ell=1}^{D}$ is a partitioning of all the states. Furthermore, S_0 contains just one state. In other words, when we reach class S_0, it means that the walkers have met. Now, we introduce a new Markov chain, by letting each class to be a single state by itself, and we denote it by S_ℓ. Finally, we assign the following transition probabilities to the new Markov chain. For each $\ell = 1,2,\ldots,D$, let

1. $\mathbb{P}\{S_\ell \to S_{\ell-1}\} = \min_{x\in S_\ell, y\in S_{\ell-1}} \mathbb{P}\{x \to y\}$,
2. $\mathbb{P}\{S_\ell \to S_\ell\} = \min_{x\in S_\ell} \mathbb{P}\{x \to x\}$,
3. $\mathbb{P}\{S_\ell \to S_{\ell+1}\} = 1 - \min_{x\in S_\ell} \mathbb{P}\{x \to x\} - \min_{x\in S_\ell, y\in S_{\ell-1}} \mathbb{P}\{x \to y\}$.

Also, note that $\mathbb{P}\{S_\ell \to S_{\ell+1}\} \geq 0$, and hence the above transition probabilities are well defined. In fact, assigning the above transition probabilities for the new chain is based on a worst case scenario which keeps the walkers away from each other for the longest period of time. In other words, this new birth-and-death chain slows down the progress of moving the walkers toward each other. As an example, given that the distance of the walkers at the current time instant is ℓ, i.e., $d_{\mathcal{G}}(x^{(0)}, x^{(2)}) = \ell$ (and hence $x \in S_\ell$), the probability that at the next time instant the walkers will get closer to each other in the original process (in the new birth-and-death chain this means that x moves from S_ℓ to $S_{\ell-1}$) is at least as large as that in the new birth-and-death chain. Therefore, it is not hard to see that the probability that the walkers in the original process meet over every sample path is at least as large as the probability that the equivalent associated sample path in the new birth-and-death process hits the class S_0. Hence, the expected time to hit the state S_0 is always an upper bound for meeting time in the original coupled process. Finally, we note that since in the original process each edge is chosen with probability $\frac{1}{m}$, the above assigned probabilities cannot be smaller than $\frac{1}{m}$. □

Corollary A.1 *Assume that $\mathcal{G}$ is a cycle of n nodes. Then, $\overline{T} \leq \frac{n(n-1)(n-3)}{16} + \frac{2n+1}{2}$.*

Proof Analyzing the birth-and-death chain described in Lemma A.2 for a cycle with n nodes, we can bound $\overline{T}$ from above. For such a graph, the new Markov chain has the structure shown in Fig. A.2.

Therefore, $\overline{T}$ is bounded from above by $H(S_{[\frac{n-1}{2}]}, S_0)$ in the birth-and-death diagram of Fig. A.2. A simple calculation shows that $H(S_{[\frac{n-1}{2}]}, S_0) = \frac{n(n-1)(n-3)}{16} + \frac{2n+1}{2}$ and the result follows. □

Corollary A.2 *If $\mathcal{G}$ is a line graph with n nodes, then $\overline{T} \leq \frac{(n-1)^2(n+1)}{4}$.*

Proof This follows from a similar argument as in the proof of Corollary A.1, and the bound coincides with the result given in [3]. □

Corollary A.3 *For a line graph and cycle with n nodes, we have $C_1 n^3 \leq \overline{T} \leq C_2 n^3$, where $0 < C_1 < C_2$ are two constants.*

Proof We prove the result for the line graph; for the cycle graph the proof is similar. From (2.1), since the degree of each node is at most 2, we have

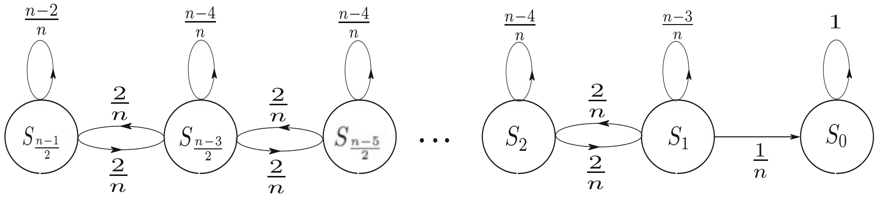


Fig. A.2 Birth-and-death chain for a cycle with n nodes

$$2\sum_{i}\left[\mathcal{R}(x \leftrightarrow y)+\mathcal{R}(y \leftrightarrow i)-\mathcal{R}(x \leftrightarrow i)\right] \geq H(x, y),$$

where $H(\cdot, \cdot)$ is the expected hitting time function of the simple random walk. Replacing this inequality in (3.13), and since $m = n - 1$, we get $H_{\mathcal{Z}}(x, y) \geq \frac{n+3}{4} H(x, y)$. Also, using Lemma 3.2 we can write

$$\overline{T} \geq \frac{1}{2} H_{\mathcal{Z}} \geq \frac{n+3}{8} \max_{x,y} H(x, y) = \frac{n+3}{8}(n-1)^2.$$

This relation in view of Corollary A.2 completes the proof. □

References

1. W. N. Anderson Jr and T. D. Morley, "Eigenvalues of the Laplacian of a graph*," *Linear and Multilinear Algebra*, vol. 18, no. 2, pp. 141–145, 1985.
2. L. Jiong-Sheng and Z. Xiao-Dong, "A new upper bound for eigenvalues of the Laplacian matrix of a graph," *Linear Algebra and Its Applications*, vol. 265, no. 1, pp. 93–100, 1997.
3. A. Kashyap, T. Başar, and R. Srikant, "Quantized consensus," *Automatica*, vol. 43, no. 7, pp. 1192–1203, 2007.

Appendix B
Resistance-Based Diffusion Game

In this appendix, we formulate the competitive diffusion game as a zero-sum game, and from a new point of view. In particular, we interpret the diffusion process on the network as a evolution of a Markov chain. Using some analogy between electric circuits and reversible Markov chains, we formulate the utility of the players using the effective resistances of an appropriate electric circuit.

In the 2-player diffusion game introduced in Sect. 5.1, assuming that players A and B place their seeds at a and b, respectively, a diffusion process unfolds on the network such that each node adopts either of the two types or remains uninfected or changes to gray. In particular, an uninfected node will adopt one type rather than the other (let us say type A), if the diffusion process initiated from a reaches earlier to that particular uninfected node than the diffusion process initiated from node b. Inspired from this observation, we look at the adoption process from another angle. In other words, we look at each uninfected node as a node that is searching to adopt one of the types, and a natural question is: What is the likelihood that an uninfected node adopts type A rather than type B, and vice versa? To address this question, we use a Markov chain interpretation of the diffusion process. Note that in this new formulation we eliminate the situations where the nodes can change to gray or remain uninfected. As an example, one can think of the cell phone market where every individual will eventually choose to buy one of the brands instead of not having access to cell phone at all.

Let us assume that the two players A and B place their seeds at nodes a and b, respectively. For a given arbitrary node x in the network, we may assume that node x runs a simple random walk in order to search one of the types around him. In this fashion, the probability that node x adopts type A is equal to the probability that node x finds type A before type B, which can be seen as the probability that the simple random walk initiated at node x hits node a before b. We denote this probability by $\mathbb{P}\{\tau_a^x < \tau_b^x\}$. Therefore, assuming that each uninfected node has total benefit of 1 for the players, the share of player A from winning node x is $\mathbb{P}\{\tau_a^x < \tau_b^x\}$, and similarly the share of player B from node x will be $\mathbb{P}\{\tau_b^x < \tau_a^x\} = 1 - \mathbb{P}\{\tau_a^x < \tau_b^x\}$. Therefore, the utility of Players A and B for initial seed placement (a, b) can be defined as

S.R. Etesami, *Potential-Based Analysis of Social, Communication, and Distributed Networks*, Springer Theses, DOI 10.1007/978-3-319-54289-8

$$\begin{aligned} U_A(a,b) &= \sum_{x\in V(\mathcal{G})} \mathbb{P}\{\tau_a^x < \tau_b^x\} \\ U_B(a,b) &= \sum_{x\in V(\mathcal{G})} \mathbb{P}\{\tau_b^x < \tau_a^x\}. \end{aligned} \tag{B.1}$$

Next, in order to formulate the utility functions defined in (B.1) based on effective resistances of an appropriate network, we consider the following result from the theory of Markov chains.

Lemma B.1 *For a random walk on a network and for every three vertices a, b and x, we have*

$$\mathbb{P}\{\tau_a^x < \tau_b^x\} = \frac{\mathcal{R}(a \leftrightarrow b) + \mathcal{R}(b \leftrightarrow x) - \mathcal{R}(x \leftrightarrow a)}{2\mathcal{R}(a \leftrightarrow b)}.$$

Proof See [1, Chap. 10]. □

Now, combining Lemma B.1 with equation (B.1), one can formulate the utility of the first player (type A) as

$$\begin{aligned} U_A(a,b) &= \sum_{x\in V(\mathcal{G})} \frac{\mathcal{R}(a \leftrightarrow b) + \mathcal{R}(b \leftrightarrow x) - \mathcal{R}(x \leftrightarrow a)}{2\mathcal{R}(a \leftrightarrow b)} \\ &= \frac{n}{2} + \frac{1}{2\mathcal{R}(a \leftrightarrow b)} \sum_{x\in V(\mathcal{G})} \big(\mathcal{R}(b \leftrightarrow x) - \mathcal{R}(x \leftrightarrow a)\big) \\ &= \frac{n}{2} + \frac{(\mathbf{e}_b - \mathbf{e}_a)'\mathbf{R}\mathbf{1}}{2\mathcal{R}(a \leftrightarrow b)}, \end{aligned} \tag{B.2}$$

where in the last equality $\mathbf{R}$ denotes the effective resistance matrix of the network, i.e., $\mathbf{R}_{ij} = \mathcal{R}(i \leftrightarrow j)$. Similarly, the utility of the second player (type B) can be expressed as

$$U_B(a,b) = \frac{n}{2} + \frac{(\mathbf{e}_a - \mathbf{e}_b)'\mathbf{R}\mathbf{1}}{2\mathcal{R}(a \leftrightarrow b)}. \tag{B.3}$$

From (B.2) and (B.3), it is not hard to see that $U_A(a,b)+U_B(a,b) = n$, which means that the above utility functions define a constant-sum game (strategically equivalent to a zero-sum game) over the network $\mathcal{G}$. In fact, defining the utility functions as above captures some of the hidden correlations between the players due to the complex structure of the network. Furthermore, such formulations allow us to generalize the competitive diffusion game to networks where edges have some positive (not-unity) weights. In this case, the resistance of the edges can be defined to be some positive constant other than 1. More information on the connection between random walks over weighted graphs and electric circuits can be found in [1].

Extension of Resistance-Based Diffusion Game

By taking a closer look at the utility functions given in (B.2) and (B.3), one can readily foresee extensions to the case where the players are allowed to choose a subset of nodes instead of only one node as their initial seed placement. In such a case, assuming that players choose $S_A \subset V(\mathcal{G})$ and $S_B \subset V(\mathcal{G})$ as their initial seeds, one can consider the hitting time of an uninfected node x to these subsets, i.e., $\tau^x_{S_A}$ and $\tau^x_{S_B}$. For this purpose, one can construct a new network $\mathcal{G}_{new}$ by contracting all the nodes in S_A and S_B into two new nodes a and b, respectively, such that a node $x \in V(\mathcal{G}) \setminus \{S_A \cup S_B\}$ in the new network $\mathcal{G}_{new}$ is connected to a or b, respectively, if and only if x was connected to at least one of the nodes in the sets S_A or S_B in the original network $\mathcal{G}$. Therefore, it is not hard to see that $\tau^x_{S_A}$ and $\tau^x_{S_B}$ in the original network are equal to τ^x_a and τ^x_b in the new network, respectively. In particular, forming the resistance matrix $\mathbf{R}_{new}$ for the new network $\mathcal{G}_{new}$, it is possible to come up with expressions similar to (B.2) and (B.3) as the utility functions of the players based on the effective resistances of the new network $\mathcal{G}_{new}$. At this point, we note that the utilities as above allow the players to compute their payoffs quickly in practice. In fact, knowing the structure of the network, each player can construct an electric circuit equivalent to such a network and form the effective resistance matrix in order to compute his or her utility based on (B.2) or (B.3).

Finally, we note that the resistance-based diffusion game can be extended to games with more than two players. Denoting the set of players and initial seed sets by i and S_i, $i = 1, 2, \ldots, m$, one can simply define the utility of the ith player to be

$$U_i(S_i, S_{-i}) = \sum_{x \in V(\mathcal{G})} \mathbb{P}\{\tau^x_{S_i} < \tau^x_{\cup S_{-i}}\}, \tag{B.4}$$

where $\cup S_{-i} = \cup_{k \neq i} S_k$.

Sub-modularity of Resistance-Based Diffusion Game

Unlike the diffusion game model introduced in Chap. 7, here we show the sub-modular property of the players' utility functions in the resistance-based diffusion game.

Theorem B.1 *The utility of players in the resistance-based diffusion game is a sub-modular function of their initial seed placement sets.*

Proof Given three sets A, A', B such that $A \subseteq A'$ and an arbitrary but fixed $v \in V$, we first show that

$$\mathbb{1}_{\left\{\tau^x_{A\cup\{v\}} < \tau^x_B\right\}} - \mathbb{1}_{\{\tau^x_A < \tau^x_B\}} \geq \mathbb{1}_{\left\{\tau^x_{A'\cup\{v\}} < \tau^x_B\right\}} - \mathbb{1}_{\{\tau^x_{A'} < \tau^x_B\}}. \tag{B.5}$$

To see this, we note that both sides of the above inequality are either 0 or 1. Moreover, if the right-hand side is equal to 1, then regardless of the value in the left-hand side, the right-hand side will always be greater than or equal to the left-hand side. Therefore, we need to show (B.5) only for the following two cases:

- $\mathbb{1}_{\left\{\tau^x_{A\cup\{v\}}<\tau^x_B\right\}} = \mathbb{1}_{\{\tau^x_A<\tau^x_B\}} = 1$. This means that for a specific sample path of the simple random walk originating from x, the walk hits the set A before the set B. Since $A \subseteq A' \subseteq A' \cup \{v\}$, therefore, such a sample path must hit the sets A' and $A' \cup \{v\}$ before the set B as well, i.e., $\mathbb{1}_{\left\{\tau^x_{A'\cup\{v\}}<\tau^x_B\right\}} = \mathbb{1}_{\{\tau^x_{A'}<\tau^x_B\}} = 1$. Therefore, in this case both sides of the relation (B.5) are equal to zero and hence the inequality holds.
- $\mathbb{1}_{\left\{\tau^x_{A\cup\{v\}}<\tau^x_B\right\}} = \mathbb{1}_{\{\tau^x_A<\tau^x_B\}} = 0$. This situation occurs when the simple random walk initiated at node x hits the set B before the set $A \cup \{v\}$. In particular, such a random walk must hit the set B before hitting the node $\{v\}$. Now if $\mathbb{1}_{\left\{\tau^x_{A'\cup\{v\}}<\tau^x_B\right\}} = 0$, then the right-hand side of (B.5) will be zero and the inequality clearly holds. Otherwise, if $\mathbb{1}_{\left\{\tau^x_{A'\cup\{v\}}<\tau^x_B\right\}} = 1$, and since we already know that the random walk does not hit the node v before the set B, thus it must hit the set A' before the set B, i.e., $\mathbb{1}_{\{\tau^x_{A'}<\tau^x_B\}} = 1$. Therefore, in this case the right-hand side of (B.5) will be equal to zero and hence the inequality holds.

Now by taking expectation of both sides of (B.5) we get

$$\mathbb{P}\left\{\tau^x_{A\cup\{v\}} < \tau^x_B\right\} - \mathbb{P}\left\{\tau^x_A < \tau^x_B\right\} \geq \mathbb{P}\left\{\tau^x_{A'\cup\{v\}} < \tau^x_B\right\} - \mathbb{P}\left\{\tau^x_{A'} < \tau^x_B\right\}. \tag{B.6}$$

Finally, given an initial strategy profile (S_i, S_{-i}) an arbitrary but fixed $v \in V$, and every set $S'_i \supseteq S_i$ of V, by replacing, respectively, S_i, S'_i and $\cup S_{-i}$ instead of A, A' and B in (B.6), summing over all $x \in V$ and in view of the definition of utility functions given in (B.4), we obtain $U_i(S_i \cup \{v\}, S_{-i}) - U_i(S_i, S_{-i}) \geq U_i(S'_i \cup \{v\}, S_{-i}) - U_i(S'_i, S_{-i})$. □

In fact, Theorem B.1 tells us that by applying a simple greedy algorithm, each player can find its optimum seed placement up to a constant factor of $1 - \frac{1}{e}$. This simply follows from applying the result of greedy algorithm for sub-modular cost functions [2].

References

1. D. A. Levin, Y. Peres, and E. L. Wilmer, *Markov Chains and Mixing Times*. American Mathematical Society, 2009.
2. E. Mossel and S. Roch, "Submodularity of influence in social networks: From local to global," *SIAM Journal on Computing*, vol. 39, no. 6, pp. 2176–2188, 2010.

Index

S.R. Etesami, *Potential-Based Analysis of Social, Communication, and Distributed Networks*, Springer Theses, DOI 10.1007/978-3-319-54289-8